MULTISOURCE CROSSTALK REDUCTION AND PLANE-WAVE LEAST-SQUARES KIRCHHOFF MIGRATION

by

Xin Wang

A dissertation submitted to the faculty of
The University of Utah
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Geophysics

Department of Geology and Geophysics

The University of Utah

August 2014

The University of Utah Graduate School

STATEMENT OF DISSERTATION APPROVAL

This dissertation of **Xin Wang**

has been approved by the following supervisory committee members:

Gerard T. Schuster	, Chair	**05/09/2014** Date Approved
Michael S. Thorne	, Member	**05/09/2014** Date Approved
Ronald L. Bruhn	, Member	**05/23/2014** Date Approved
Sergio Chavez-Perez	, Member	**05/09/2014** Date Approved
Sherif Hanafy	, Member	**05/18/2014** Date Approved

and by **John M. Bartley**, Chair of

the Department of **Geology and Geophysics**

and by David B. Kieda, Dean of the Graduate School.

ABSTRACT

Least-squares migration (LSM) can produce high quality migration images from seismic data, but with high computational cost. To reduce this cost, phase-encoding techniques have been introduced, but at the expense of crosstalk noise in the image. In this dissertation, a multisource least-squares algorithm (MLSM) is proposed to eliminate the crosstalk noise and also significantly increase computational efficiency by applying plane-wave encoding to the data. I denote variations of this method as plane-wave least-squares migration (PWLSM) and regularized plane-wave least-squares migration (RPWLSM).

There are four main chapters in this dissertation.

In Chapter 2, the theory of multisource crosstalk reduction is presented and numerically validated, which provides the basis for Kirchhoff MLSM and PWLSM. Numerical results suggest that: (1) combining polarity and time statics produces a better migration image than computed with either polarity or time statics alone; (2) MLSM achieves the best crosstalk reduction when the encoding function varies at each iteration.

In Chapter 3, MLSM is used with Kirchhoff migration for a marine acquisition geometry. Application of wave-equation MLSM to marine streamer data is typically hampered by the variable location of receivers for any shot. However, this problem does not exist for Kirchhoff MLSM, which can be applied to marine streamer data without any restrictions on the acquisition geometry. Kirchhoff MLSM achieves the best performance by changing the encoding function at each iteration. MLSM achieves a significant savings in IO and memory costs, but does not reduce the CPU cost, which suggests a better phase-encoding technique is needed.

In Chapter 4, a 2D plane-wave encoding technique is developed for Kirchhoff LSM to produce high quality images (compared to standard Kirchhoff migration) with a significant gain in computational efficiency (compared to Kirchhoff LSM). A finite-difference solver for the eikonal equation is developed for calculation of plane-wave travel times. To remedy the high sensitivity of PWLSM to the migration velocity accuracy, RPWLSM is developed where each plane-wave is migrated to give its own migration image, hence it increases the robustness of this algorithm when the migration velocity model contains bulk errors. By

noting that images from different plane-wave gathers with slightly different encoding parameters are similar, a regularization term is adopted to encourage this similarity. Numerical tests on synthetic and field data are performed to illustrate the benefits and limitations of PWLSM and RPWLSM.

In Chapter 5, the proposed algorithms of PWLSM and RPWLSM are extended from 2D to 3D. For 3D narrow azimuth streamer geometries an inline only plane-wave encoding strategy (also known as the cylindrical-wave encoding) is adopted. Cylindrical-wave least-squares migration (CWLSM) and regularized cylindrical-wave least-squares migration (RCWLSM) methods are tested on synthetic and field data. Results show the improved quality of 3D PWLSM and CWLSM images compared to standard Kirchhoff migration with an acceptable computational cost.

To my family.

CONTENTS

ABSTRACT ..... **iii**

LIST OF FIGURES ..... **viii**

LIST OF TABLES ..... **xiv**

ACKNOWLEDGMENTS ..... **xv**

CHAPTERS

1. INTRODUCTION AND OVERVIEW **1**

1.1 Chapter 2: Multisource Crosstalk Reduction 3
1.2 Chapter 3: Multisource Least-squares Kirchhoff Migration 3
1.3 Chapter 4: 2D Plane-wave Least-squares Kirchhoff Migration 3
1.4 Chapter 5: 3D Plane-wave Least-squares Kirchhoff Migration 4
1.5 Technical Contribution in this Dissertation 6

2. MULTISOURCE CROSSTALK NOISE REDUCTION **8**

2.1 Introduction 8
2.2 Phase-Encoded Migration of Supergathers 10
2.3 Iterative Phase-Encoded Migration of Supergathers 14
2.3.1 Iterative Stacking of Encoded Migration Images 14
2.3.2 Iterative Least-squares Migration 14
2.4 Signal-to-Noise Ratio vs Number of Blended Shot Gathers 17
2.4.1 Theory 17
2.4.2 Numerical Tests 18
2.5 Ensemble Averaging and Crosstalk Noise 21
2.5.1 Source Time Statics 21
2.5.2 Polarity Statics 23
2.6 Static, Hybrid and Dynamic Phase-encoding 30
2.6.1 Theory 30
2.6.2 Numerical Tests 31
2.7 Discussion and Conclusions 31

3. MULTISOURCE LEAST-SQUARES KIRCHHOFF MIGRATION **34**

3.1 Introduction 34
3.2 Theory 35
3.2.1 Kirchhoff Modeling and Migration 35
3.2.2 Least-squares Kirchhoff Migration 36
3.2.3 Multisource Kirchhoff Modeling and Migration 37
3.2.4 Multisource Least-squares Kirchhoff Migration 37

3.3 Numerical Results 39
3.3.1 Marmousi2 Model with FD Data 39
3.3.2 Gulf of Mexico Marine Streamer Data 44
3.4 Discussion and Conclusion 51

4. 2D PLANE-WAVE LEAST-SQUARES KIRCHHOFF MIGRATION . . 52

4.1 Introduction 52
4.2 Theory 54
4.2.1 2D Plane-wave Encoding 54
4.2.2 Plane-wave Kirchhoff Modeling and Migration 54
4.2.3 Plane-wave Least-squares Migration 56
4.2.4 Regularize Plane-wave Least-squares Migration 58
4.2.5 Plane-wave Eikonal Solver 59
4.3 Numerical Tests 67
4.3.1 Synthetic Data Test 67
4.3.2 Field Data Test 74
4.4 Conclusions 79

5. 3D PLANE-WAVE LEAST-SQUARES KIRCHHOFF MIGRATION . . 82

5.1 Introduction 82
5.2 Theory 83
5.2.1 3D Plane-wave Encoding 83
5.2.2 3D Plane-wave Kirchhoff Modeling and Migration 83
5.2.3 3D Plane-wave Least-squares Kirchhoff Migration 85
5.2.4 3D Regularized Plane-wave Least-squares Migration 86
5.2.5 Cylindrical-wave Encoding 87
5.2.6 Cylindrical-wave Kirchhoff Modeling, Migration and Least-squares Migration 87
5.2.7 3D Plane- and Cylindrical-waves Eikonal Solvers 90
5.3 Numerical Tests 91
5.3.1 Synthetic Test of 3D Plane-wave Data 91
5.3.2 Synthetic Test of 3D Cylindrical-wave Data 98
5.3.3 Field Test of 3D Cylindrical-wave Data 103
5.4 Conclusions 103

APPENDICES

A. TRIM STATICS AND LSM 108

B. 3D FINITE DIFFERENCE STENCILS FOR EIKONAL SOLVER 111

REFERENCES 113

LIST OF FIGURES

1.1 Schematic diagrams of the advantages of MLSM and PWLSM compared to standard LSM: a) the acquisition geometry with N_S shots with each shot shooting into N_G receivers, b) single common shot gathers, c) supergathers by multisource phase-encoding, d) plane-wave gathers and e) LSM, MLSM and PWLSM achieve the same quality images. By transforming single CSGs to plane-wave gathers, all the IO, memory and CPU costs with PWLSM are decreased to $\frac{N_P \times N_G}{N_S \times N_G} = \frac{N_P}{N_S}$ relative to LSM. 2

1.2 Schematic diagrams of the plane-wave transform: a) the 2-layer velocity model and 9 shots shooting into 9 receivers, b) common shot gathers, c) linear time shifting, d) summation of the shifted traces, and e) 5 plane-wave gathers. 5

2.1 Blended shot migration: a) Time-shifted shot gathers, b) blended supergather created by blending and stacking S time-shifted shot gathers, c) migration images after migrating the supergather for each shot position and d) final image after summing S migration images. The final SNR enhancement is $\sqrt{G}$. 12

2.2 Phase-encoded iterative migration images for the SEG salt model: a) standard migration, b) one supergather migration and c-h) iterative stacking images after 30, 60, 90, 120, 150 and 180 iterations. Here the standard deviation of the source statics is 0.05 second. 19

2.3 The solid (dotted) line indicates the measured SNR for phase-encoded iterative stacking migration, where the input data set consists of a 320-shot supergather associated with the SEG/EAGE salt model; the dashed line indicates the predicted relation from equation 2.21. Both curves have been normalized to the value 1 for the first iterate. 20

2.4 Iterative stacking migration images for data associated with the point scatterer model and the 5 Hz narrow band source wavelet: a) standard migration, b) - f) show the iterative stacking images for the standard deviation of source static of 0.01 s, 0.03 s, 0.05 s, 0.08 s and 0.1 s. Only 100 stacks were needed to compute these images. 24

2.5 The solid line with diamonds indicates the measured normalized crosstalk damping coefficients for phase-encoded iterative stacking migration, where the input data set consists of a 256-shot supergather associated for a point scatterer model with a homogeneous background velocity of 2000 m/s; the dashed line indicates the predicted relation from equation 2.21, and the input source wavelet is band limited with a dominant frequency of 5 Hz. The observed curve is nearly the same for a range of τ_s standard deviations between 0.01 s and 0.1 s. 25

2.6 Iterative stacking migration images for the SEG salt model, where the narrow band source wavelet is peaked at 50 Hz: a) standard migration, b), c), and d), iterative stacking images after 1, 50, 500 iterations. Here, the standard deviation of the source statics is 0.05 second. 25

2.7 The solid lines with different colors indicate the measured normalized crosstalk damping coefficients for phase-encoded iterative stacking migration after 500, 1000, 1500, 2000 2500, and 3000 stacks, where the input data consists of a 320-shot supergather associated with the SEG salt model. The dashed line indicates the predicted relation from equation 2.24, where the input narrow band source wavelet is peaked at 50 Hz. 26

2.8 Comparison of iterative migration images $\mathbf{m_{enc}}$ with: a1) time statics encoding ($\sigma = 0.1$ s), b1) source polarity encoding and c1) the combination of both. b1), b2) and b3) show the residual error $\boldsymbol{\delta}$, i.e., the difference between the standard single-source migration $\hat{\mathbf{m}}$ and $\mathbf{m_{enc}}$, respectively. All iterative migration images are after 3000 stacks. The residual error is scaled by a factor of 20, for ease of perception, in b2) and c2). The input data set consists of a blended 320-shot supergather associated with the SEG salt model, and the input source wavelet is a Ricker wavelet with peak frequency at 50 Hz. 28

2.9 SNR vs the standard deviation of time statics, obtained from iterative migration after 3000 stacks, using encoding techniques such as: (bottom curve, solid, with diamonds) time statics, (middle line, solid) polarity, and (top curve, dashed, with diamonds) combined time statics and polarity. The input data set consists of a blended 320-shot supergather associated with the SEG salt model, and the input source wavelet is a Ricker wavelet with peak frequency at 50 Hz. 29

2.10 Comparison of standard, static phase-encoded and dynamic phase-encoded LSM images: a) through d) show the standard LSM images after 5, 30, 80 and 300 iterations; e) through h) illustrate the static phase-encoded LSM images after 5, 30, 80 and 300 iterations of a 320-shot supergather; while i) through l) present the dynamic phase encoding LSM images after 5, 30, 80 and 300 iterations of a 320-shot supergather. All input data are associated with the SEG/EAGE salt model and a source Ricker wavelet with a peak frequency of 50 Hz. 32

2.11 The solid line indicates the measured relative model error of least-squares migration with dynamic phase-encoding source time shift, while the dashed line indicates the one with static phase-encoding source time shift. Input data set consists of a 320-shot supergather associated with the SEG salt model, the input source wavelet is a Ricker wavelet with the peak frequency at 50 Hz. . . . 33

3.1 Steps for creating a phase-encoded supergather and decoding. (a) shows the receiver and shot positions associated with two distinct shot gathers; (b), (d) and (f) show the original, polarity encoded and receiver-side time shifted encoded common shot gather (CSG) 1, while (c), (e) and (g) are for CSG 2; (h) is the supergather after summing of (f) and (g); and (i) and (j) are the decoded CSGs for migration. 38

3.2 Illustration of steps for multisource Kirchhoff migration. (a) shows the single shot CSGs with a marine streamer acquisition aperture, receiver positions are different for each shot, (b) shows phase-encoding which shifts CSG1 1 second later, (c) shows the supergather data generated by stacking traces from CSG1 and CSG2, (d) shows the decoded shot, which shifts the supergather trace 1 second back and (e) shows the migration of the decoded CSGs with the associated source and receiver positions. 38

3.3 2D Marmousi2 model: a) velocity model and b) reflectivity model. Synthetic data are generated by a FD solution to the acoustic wave equation with a Ricker wavelet at the peak frequency of 20 Hz. 40

3.4 Standard Kirchhoff migration image of the 2D Marmousi2 model. 40

3.5 Least-squares migration image after 15 iterations of the 2D Marmousi2 model. 41

3.6 Zoom views of red boxes in Figures 3.3-3.5: a) velocity model, b) reflectivity model, c) Kirchhoff migration and d) least-squares migration. 41

3.7 Zoom views of yellow boxes in Figures 3.3-3.5: a) velocity model, b) reflectivity model, c) Kirchhoff migration and d) least-squares migration. 42

3.8 Zoom views of blue boxes in Figures 3.3-3.5: a) velocity model, b) reflectivity model, c) Kirchhoff migration and d) least-squares migration. 42

3.9 Multisource least-squares migration image with static phase encoding after 20 iterations. 43

3.10 Multisource least-squares migration image with hybrid phase encoding after 20 iterations. 43

3.11 Multisource least-squares migration image with dynamic phase encoding after 20 iterations. 45

3.12 Zoom views of red boxes in Figures 3.5, 3.9, 3.10 and 3.11: a) standard LSM, b) MLSM with static encoding, c) MLSM with hybrid encoding and d) MLSM with dynamic encoding. 45

3.13 Zoom views of yellow boxes in Figures 3.5, 3.9, 3.10 and 3.11: a) standard LSM, b) MLSM with static encoding, c) MLSM with hybrid encoding and d) MLSM with dynamic encoding. 46

3.14 Zoom views of blue boxes in Figures 3.5, 3.9, 3.10 and 3.11: a) standard LSM, b) MLSM with static encoding, c) MLSM with hybrid encoding and d) MLSM with dynamic encoding. 46

3.15 P-wave velocity model estimated of the Gulf of Mexico marine data from full waveform inversion. 47

3.16 Standard Kirchhoff migration image of GOM marine data. 48

3.17 Standard least-squares Kirchhoff migration image of GOM marine data after 30 iterations. 48

3.18 Zoom views of blue boxes in Figures 3.16-3.21: a) standard KM, b) standard LSM, c) MLSM with static encoding and d) MLSM with dynamic encoding. . 49

3.19 Zoom views of yellow boxes in Figures 3.16-3.21: a) standard KM, b) standard LSM, c) MLSM with static encoding and d) MLSM with dynamic encoding. . 49

3.20 Multisource least-squares migration image with static phase-encoding of GOM marine data after 30 iterations. 50

3.21 Multisource least-squares migration image with dynamic phase-encoding of GOM marine data after 30 iterations. 50

4.1 2D plane-wave encoding, where the time shift is a linear function with respect to the source location x_s, reference source location x_0 and the slope is the ray parameter p. 55

4.2 PWKM images on the Marmousi2 model: a) single plane-wave image, b) and c) are the stacked images computed from 51 and 11 plane-waves images, respectively. 57

4.3 Gridpoint models for the 2D FD plane-wave eikonal solver: a) the current expanding wavefront with N known points, b) the next expanding wavefront with N+3 known points, c)-e) three different FD stencils. Green-filled-circles indicate the *global minimum* on the wavefront, red-filled-circles indicate all the other parts of the timed wavefront, and the black-empty-circles with green squares indicate the points that will be expanded. 61

4.4 The four plane-wave travel time contours associated with the ray parameters 0.4, 0.2, 0 and -0.1 ms/m, respectively. 63

4.5 The six plane-wave snapshots at t 0.22, 0.42, 0.72, 1.02, 1.32 and 1.62 s, respectively. Snapshots are computed by a FD solution to the acoustic wave equation with plane-wave source of $p = 0.2$ ms/m. 63

4.6 The six contours of the eikonal travel times superimposed on the snapshots computed by a FD solution to the wave equation. 64

4.7 Marmousi2 model used for testing the plane-wave eikonal solver. 64

4.8 The four plane-wave travel time contours associated with the ray parameters 0.4, 0.2, 0 and -0.1 ms/m, respectively. 65

4.9 The six plane-wave snapshots at t 0.17, 0.32, 0.47, 0.62, 0.77 and 0.92 s, respectively. Snapshots are computed by a FD solution to the acoustic wave equation with plane-wave source of $p = 0.2$ ms/m. 65

4.10 The six contours of the eikonal travel times superimposed on the snapshots computed by a FD solution to the wave equation. 66

4.11 Marmousi2 velocity models: a) true velocity, b) smoothed migration velocity, c) erroneous velocity for sensitivity test, d) difference between b) and c). The velocity error will shift reflector positions by 2 and 2.5 wavelengths at the respective depths of 1.5 and 2 km. 68

4.12 Plane-wave gathers with ray parameters of a) -0.279 and b) 0.043 ms/m. 69

4.13 CSG KM and PWKM images computed with the true velocity model: a) CSM KM and b) PWKM images. 70

4.14 LSM images with accurate velocity model after 10 iterations: a) plane-wave least-squares migration and b) regularized plane-wave least-squares migration. 71

4.15 Convergent curves of PWLSM and RPWLSM for the true and erroneous migration velocities. Black lines are for the true migration velocity and red lines are the for the erroneous velocity model. 71

4.16 Migration images with erroneous velocity model for: a) PWKM, b) PWLSM and c) RPWLSM after 10 iterations. 72

4.17 CIGs with erroneous velocity of: (a) PWKM and (b) RPWLSM after 10 iterations. 73

4.18 CSG and PW gathers: a)-b) two CSGs and c-d) two plane-wave gathers. 75

4.19 Migration velocity of the 2D Marine data. 75

4.20 Images of: a) conventional KM and b) PWKM. 76

4.21 Images of: a) PWLSM, b) RPWLSM and c) RPWLSM after applying trim statics. 77

4.22 Zoom views of the blue boxes in Figures 4.20b and 4.21a-c. 78

4.23 Zoom views of the red boxes in Figures 4.20b and 4.21a-c. 78

4.24 CIGs of: a) PWKM and b) RPWLSM. 80

4.25 Images of: a) PWKM and b) RPWLSM with zero incident angle. 81

4.26 Convergent curves for PWLSM (red) and RPWLSM (black). 81

5.1 3D plane-wave encoding. 84

5.2 3D cylindrical-wave encoding for a nonzero p_y value. 88

5.3 Overthrust velocity model for testing the 3D plane-wave eikonal solver, where the model size is $5 \times 5 \times 2.5$ km with a 25 m gridpoint interval: a) XY plane at $z = 1$ km, b) XZ plane at $y = 2.5$ km and c) YZ plane at $x = 2.5$ km. 92

5.4 3D plane-wave travel times with ray parameter $p_x = -0.02$ and $p_y = 0.02$ ms/m: a) XY plane at $z = 1$ km, b) XZ plane at $y = 2.5$ km and c) YZ plane at $x = 2.5$ km. 92

5.5 3D plane-wave travel times with ray parameters $p_x = 0$ and $p_y = 0.0048$ ms/m: a) XY plane at $z = 1$ km, b) XZ plane at $y = 2.5$ km and c) YZ plane at $x = 2.5$ km. 93

5.6 Overthrust velocity for testing the 3D cylindrical-wave eikonal solver, where the model size is $1.5 \times 5 \times 2.5$ km with a 25 m gridpoint interval: a) XY plane at $z = 1$ km, b) XZ plane at $y = 2.5$ km and c) YZ plane at $x = 0.75$ km. 93

5.7 3D cylindrical-wave travel times with ray parameters $p_y = -0.015$ ms/m: a) XY plane at $z = 1$ km, b) XZ plane at $y = 2.5$ km and c) YZ plane at $x = 0.75$ km. 94

5.8 3D cylindrical-wave travel times with ray parameters $p_y = 0.026$ ms/m: a) XY plane at $z = 1$ km, b) XZ plane at $y = 2.5$ km and c) YZ plane at $x = 0.75$ km. 94

5.9 CSG-domain KM images: a) XY plane at $z = 1$ km, b) XZ plane at $y = 1.875$ km and c) YZ plane at $x = 2.5$ km. 95

5.10 PW-domain KM images: a) XY plane at $z = 1$ km, b) XZ plane at $y = 1.875$ km and c) YZ plane at $x = 2.5$ km. 95

5.11 PWLSM images: a) XY plane at $z = 1$ km, b) XZ plane at $y = 1.875$ km and c) YZ plane at $x = 2.5$ km. 96

5.12 RPWLSM images: a) XY plane at $z = 1$ km, b) XZ plane at $y = 1.875$ km and c) YZ plane at $x = 2.5$ km. 96

5.13 CIGs of different inline (y) positions at $x = 2.5$ km for: a) PWKM image and b) RPWLSM image. 97

5.14 CIGs of different crossline (x) positions at $y = 2$ km for: a) PWKM image and b) RPWLSM image. 97

5.15 Convergent curves of PWLSM and RPWLSM. Black line is for PWLSM and blue is for RPWLSM. 99

5.16 CSG-domain KM images: a) XY plane at $z = 1$ km, b) XZ plane at $y = 2.5$ km and c) YZ plane at $x = 2.5$ km. 100

5.17 Images produced by: a) CWKM, b) CWLSM and c) RCWLSM. Both CWLSM and RCWLSM are stopped after 10 iterations. 101

5.18 CIGs produced by: a) CWKM and b) RCWLSM. 102

5.19 Convergent curves of CWLSM and RCWLSM. Black line is for CWLSM and blue is for RCWLSM. 102

5.20 Velocity model of GOM data. 104

5.21 CSG KM image of GOM data. 104

5.22 CWKM image of GOM data. 105

5.23 CWLSM image of GOM data. 105

5.24 RCWLSM image of GOM data. 106

5.25 Zoom views of the blue boxes in Figures 5.22-5.24. 106

5.26 Zoom views of the green boxes in Figures 5.22-5.24 107

A.1 Comparison of conventional LSM and LSM + trim statics images after 10 iterations of LSM for 31 p-values: a) conventional LSM and b) LSM + trim statics. The data set is recorded from a marine seismic survey in the Gulf of Mexico. 110

B.1 FD stencils of: a) the wavefront at time T_{A1} propagates until the time T_{A6}, b) the wavefront at time T_{B1} propagates until the time T_{B8}, and c) the wavefront at time T_{C1} propagates until the time T_{C6}. Green-filled circles indicate the timed gridpoints, and the red-filled circles indicate the gridpoints that will be expanded. 112

LIST OF TABLES

1.1 Comparison of attributes for KM, PWKM, LSM, PWLSM and RPWLSM. Here, stand. denotes the attributes for standard KM. 7

3.1 Comparison of image quality and CPU, IO and memory costs for KM, LSM and MLSM for the Marmousi2 synthetic data test. (LSM I does not store data in memory, but LSM II does.) 47

5.1 Comparison of image quality and CPU, IO costs for KM, PWKM and PWLSM for the Overthrust synthetic data test. 99

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Gerard T. Schuster, for his guidance, support and encouragement throughout my study at the University of Utah. He taught me the in-depth understanding of geophysics theories as well as the methodology of scientific research. His passion and enthusiasm for research always encourage and inspire me during my program of study. I also wish to thank other members of my advisory committee, Dr. Ron Bruhn, Dr. Michael S. Thorne, Dr. Sherif Hanafy and Dr. Sergio Chavez-Perez for reading and correcting this dissertation and the advisory of my research.

I also thank all of my UTAM colleagues for their help and discussions. The help from Dr. Chaiwoot Boonyasiriwat, Dr. Weiping Cao, and Dr. Yanwei Xue is highly appreciated. I also enjoyed and benefited from fruitful discussion with Naoshi Aoki, Samuel Brown, Wei Dai, Shuqian Dong, Simin Huang, Shengdong Liu, Qiong Wu, and Ge Zhan. I also appreciate the efforts from all of my colleagues to make UTAM a great place for research and study.

I am very grateful for the financial support from the UTAM sponsors. I thank Aramco and Chevron for offering internships, from which I obtained industrial experiences. I'm in debt to Dr. Peeter Akerberg and Dr. Gilles Hennenfent at Chevron, Sir. Peter Pecholcs at Aramco for their guidance.

The support from computation facilities at the Center of High Performance Computation (CHPC) at the University of Utah is also highly appreciated.

CHAPTER 1

INTRODUCTION AND OVERVIEW

Least-squares migration (LSM) (Nemeth et al., 1999; Duquet et al., 2000) can generate migration images with better resolution and fewer artifacts than standard migration. However, the higher computational cost of LSM hinders its widespread application, especially for 3D seismic data. As a partial remedy, Romero et al. (2000) introduced the concept of phase-encoded migration to reduce both the size of the input data and the computation time. The main problem with this approach is that the crosstalk noise degrades the image quality. It is shown empirically (Dai et al., 2012; Huang and Schuster, 2012; Wang and Schuster, 2012) that these two shortcomings are mitigated by adapting the encoding method to LSM, which I denote as multisource least-squares migration (MLSM), to arrive at migration images comparable in quality to LSM but with sometimes less computational cost than standard migration.

In this dissertation I develop new multisource Kirchhoff migration methods that provide least-squares migration images with about the same quality as LSM but at significantly reduced cost. The key breakthrough is that I recognized that plane-wave encoding in conjunction with my development of a plane-wave eikonal solver can be combined to produce an effective and efficient LSM method. I denote this method as the plane-wave least-squares migration (PWLSM). To account for the mispositioning of reflectors due to errors in the velocity model, a regularized plane-wave least-squares Kirchhoff migration (RPWLSM) is devised so that each plane-wave gather gives rise to an individual migration image, and a regularization term is included to encourage the similarities between the migration images of similar encoding schemes. Compared to LSM, MLSM achieves better IO and memory savings but at the same CPU cost, while PWLSM and RPWLSM achieve better IO, memory and CPU cost savings. Figure 1.1 summarizes the advantages of MLSM, PWLSM and RPWLSM over LSM.

This dissertation is divided into five chapters, and the contributions from each one are summarized following.

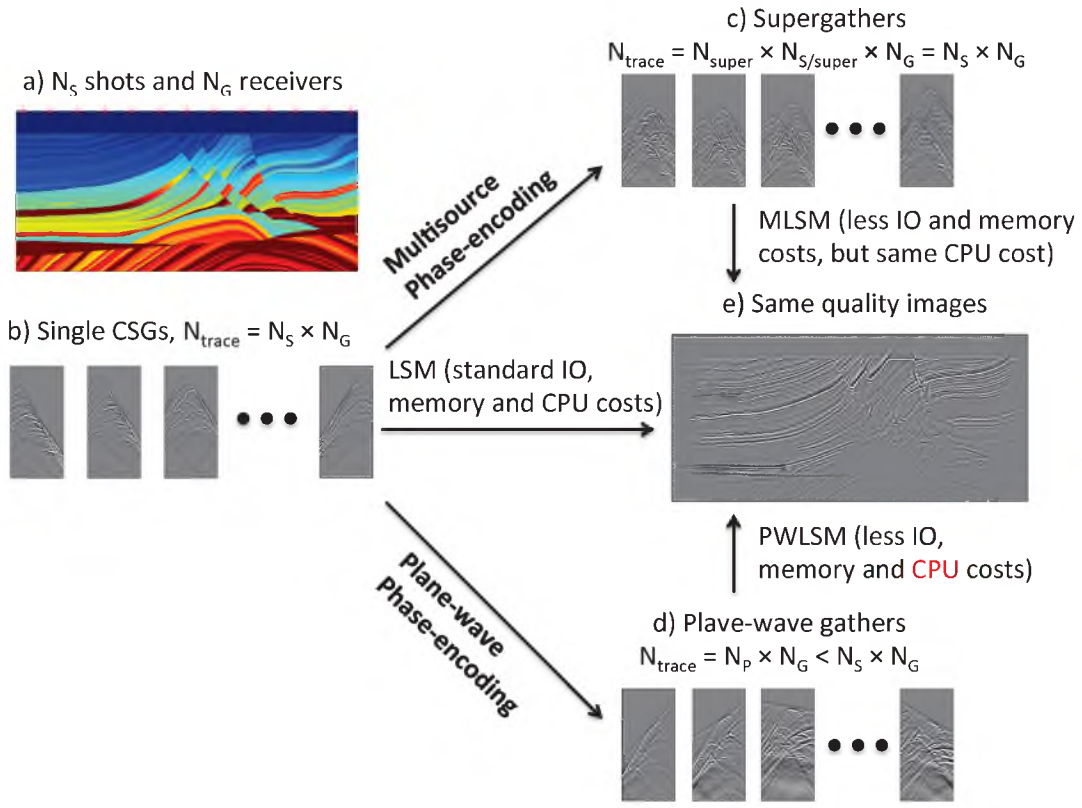

Figure 1.1. Schematic diagrams of the advantages of MLSM and PWLSM compared to standard LSM: a) the acquisition geometry with N_S shots with each shot shooting into N_G receivers, b) single common shot gathers, c) supergathers by multisource phase-encoding, d) plane-wave gathers and e) LSM, MLSM and PWLSM achieve the same quality images. By transforming single CSGs to plane-wave gathers, all the IO, memory and CPU costs with PWLSM are decreased to $\frac{N_P \times N_G}{N_S \times N_G} = \frac{N_P}{N_S}$ relative to LSM.

1.1 Chapter 2: Multisource Crosstalk Reduction

Chapter 2 presents the theory of MLSM (Schuster et al., 2011), which is the basis for developing a Kirchhoff MLSM method with plane waves. My contribution in this chapter is that I designed all of the crucial numerical experiments and carried out the simulations that validated the predictive formulas for MSLM in Schuster et al. (2011).

1.2 Chapter 3: Multisource Least-squares Kirchhoff Migration

In this chapter, MLSM proposed in Chapter 2 is used with Kirchhoff migration. Application of wave-equation MLSM to marine streamer data is typically hampered by the variable location of receivers for any shot. However, this problem does not exist for Kirchhoff MLSM, which can be applied to marine streamer data without any restrictions on the acquisition geometry. Kirchhoff MLSM with dynamic encoding achieves the best quality migration images, and this further validates the theory of crosstalk reduction for iterative least-squares migration of supergather data. My empirical results in Chapter 2 demonstrate for the first time that Kirchhoff MLSM can significantly decrease the IO and memory cost, but at the same CPU cost, compared to standard LSM.

1.3 Chapter 4: 2D Plane-wave Least-squares Kirchhoff Migration

Wave-equation LSM can achieve high computational efficiency (Dai et al., 2012; Huang and Schuster, 2012) by modeling and migrating the supergather with one finite-difference solution to the wave equation for a large distribution of encoded point sources. For Kirchhoff LSM, however, the computational cost is determined by the total number of traces, which cannot be decreased by the blended encoding of shot gathers. As a less expensive alternative, a linear time-shift phase encoding, which is identical to the tau-p transform (Zhang et al., 2005), can transfer the shot-domain data to the plane-wave domain. Figure 1.2 illustrates the procedures of plane-wave transform: (1) (as shown in Figure 1.2c), traces at the same 5th receiver position from 9 different sources are time shifted with a linear function (the shifted times are roughly indicated by the length of the dashed lines with arrows), which are related to the source positions and the ray-parameter p; (2) (as shown in Figure 1.2d), sum of all the shifted traces to form the 5th trace (as shown in the green box in Figure 1.2e) of the plane-wave gather with $p = 0$; (3), repeat (1) and (2) for all the receiver positions (orange and blue dashed lines indicate the procedures for the 1st and the 9th traces, respectively), a plane-wave gather with all the 9 traces is formed (as shown in the middle of Figure 1.2e);

(4) select different p values and repeat step (1) - (3), then 5 plane-wave gathers associated with 5 different p are formed (as shown in Figure 1.2e). By replacing the large number of shots with a smaller number of ray parameters, a significant computational savings can be achieved for plane-wave Kirchhoff migration (Akbar et al., 1996; Al-Saleh, 2002; Stoffa et al., 2006). In this chapter, I propose the new algorithm of plane-wave least-squares Kirchhoff migration (PWLSM).

For Kirchhoff migration, one of the key steps is the calculation of travel times. In this chapter, I propose a new finite-difference solver to the eikonal equation with the expanding plane-wave-front. Numerical results show that this algorithm can accurately and efficiently calculate the first-arrival plane-wave travel times.

Another disadvantage of LSM is that it is more sensitive to the accuracy of the velocity model than conventional migration. When the velocity model contains large bulk errors, the migration images from different shots are inconsistent with each other, and simple stacking will blur the image and slow the convergence. To remedy this problem, each plane-wave gather is assumed to be associated with its own reflectivity model (Dai and Schuster, 2013; Wang et al., 2013). It is also noted that the prestack images with slightly different ray parameters should be similar, so a regularized plane-wave least-squares Kirchhoff migration (RPWLSM) algorithm is proposed, where the regularization term is adapted to encourage their similarities. Also, this method provides the opportunity for migration velocity analysis and trim-statics with stacking to obtain more focused images. The computational and IO costs are significantly decreased compared to conventional LSM.

1.4 Chapter 5: 3D Plane-wave Least-squares Kirchhoff Migration

3D prestack LSM is extremely expensive in terms of CPU time, so that it has not been widely adapted in practical imaging. In this chapter, I extend the PWLSM and RPWLSM algorithms from 2D to 3D. A computational efficiency is also achieved by the 3D tau-p transform. Instead of having only one ray parameter, the 3D plane-wave encoding has two ray parameters in both the inline and crossline directions (Al-Saleh, 2002; Zhang et al., 2005; Duquet and Lailly, 2006; Chemingui et al., 2007; Vigh and Starr, 2008). For the current 3D marine streamer acquisition with a narrow azimuth geometry, the source sampling in the crossline direction is usually too sparse to support a full 3D plane-wave encoding, so an inline only plane-wave encoding is adopted. This is equivalent to a cylindrical-wave encoding (or line-source encoding). Similarly, the cylindrical-wave least-squares Kirchhoff migration (CWLSM) and regularized cylindrical-wave least-squares Kirchhoff migration (RCWLSM)

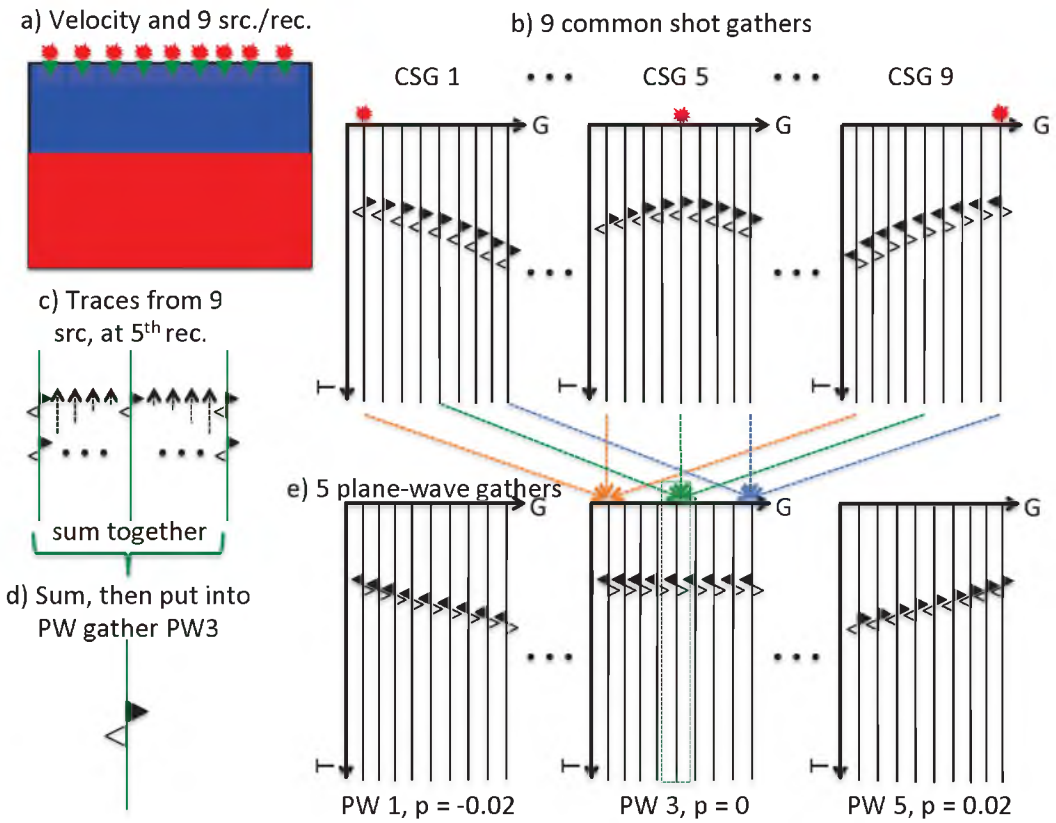

Figure 1.2. Schematic diagrams of the plane-wave transform: a) the 2-layer velocity model and 9 shots shooting into 9 receivers, b) common shot gathers, c) linear time shifting, d) summation of the shifted traces, and e) 5 plane-wave gathers.

are proposed in this chapter. The finite-difference solvers to the 3D eikonal equation with plane-wave-fronts and cylindrical-wave-fronts are also proposed to calculate the travel times. Both synthetic and field data results show the improved quality of 3D PWLSM and CWLSM images compared to standard Kirchhoff migration with an acceptable computational cost.

1.5 Technical Contribution in this Dissertation

This dissertation makes several significant contributions to the field of seismic migration.

1. In Chapter 2, the MLSM theory is presented (Schuster et al., 2011) which is the basis for developing a Kirchhoff MLSM method with plane waves. My contribution here is is that I designed all of the crucial experiments and carried out the simulations that validated the predictive formulas.

2. In Chapter 3, I apply Kirchhoff MLSM to marine steamer acquisition data. Numerical results show MLSM with dynamic encoding can achieve the same image quality as standard LSM but with a significant IO and memory savings.

3. In Chapter 4, I propose a plane-wave least-squares Kirchhoff migration which can provide high-quality images while enjoying a significant computational cost reduction by plane-wave encoding. A regularized plane-wave least-squares migration algorithm is proposed to improve the robustness when the migration velocity model contains large bulk errors, and the regularized term penalizes the differences between images of plane-wave gathers with slightly different incident angles. Also, I propose a new finite-difference solver to the eikonal equation with expanding plane-wave-fronts. Table 1.1 summarizes the computational attributes of Kirchhoff migration (KM), PWKM, LSM, PWLSM, and RPWLSM.

4. In Chapter 5, I extend the PWLSM and RPWLSM algorithms from 2D to 3D. To implement the PWLSM and RPWLSM with narrow azimuth marine streamer data, a cylindrical-wave encoding is adopted, so CWLSM and RCWLSM algorithms are introduced.

These contributions are validated by both synthetic and field data tests, and a package (called SeisF90) in FORTRAN 90 with parallel programing is developed. This package can be used as a library for related research topics in seismic imaging.

Table 1.1. Comparison of attributes for KM, PWKM, LSM, PWLSM and RPWLSM. Here, stand. denotes the attributes for standard KM.

	Image quality	**CPU expense**	**IO demand**	**Memory cost**	**Sensitivity to velocity model error**
KM	Stand.	Stand.	Stand.	Stand.	Stand.
PWKM	Stand.	Low	Low	Low	Stand.
LSM	High	Expensive	High	High	High
PWLSM	High	Acceptable	Mod.	Mod.	High
RPWLSM	High	Acceptable	Mod.	Mod.	Mod.

CHAPTER 2

MULTISOURCE CROSSTALK NOISE REDUCTION

In this chapter, formulas are derived that relate the strength of the crosstalk noise in supergather migration images to the variance of time, amplitude and polarity shifts in encoding functions (Schuster et al., 2011). A supergather migration image is computed by migrating an encoded supergather, where the supergather is formed by stacking a large number of encoded shot gathers. Analysis reveals that for the time shift static in each shot gather, the crosstalk noise is exponentially reduced with increasing variance of the static shift and the square of source frequency. Analysis also reveals that combining both polarity and time statics is a superior encoding strategy compared to using either polarity statics or time statics alone. The results in this chapter are partly published in Schuster et al. (2011) and my contributions are that I devised and computed all of the numerical simulations to validate these formulas.

To test the validity of the above formulas, iterative multisource least-squares migration (MLSM) is applied to supergathers, where the encoding function is changed at each iteration. Results empirically verify the predictions from these formulas.

2.1 Introduction

Romero et al. (2000) introduced the concept of phase-encoded migration of seismic data, where reverse time migration of blended shot gathers associated with many shot positions could be performed with just two finite-difference (FD) solutions[1] to the wave equation. Blending is performed by stacking N distinct shot gathers together to form a supergather, where each shot gather is encoded with unique phase-encoding functions ideally uncorrelated

[1] One solution is used to back-propagate the blended reflections and the other is to model the direct waves from the multiple sources, each with a different encoding function.

with each other. The potential benefit of this approach, compared to standard migration, is a N-fold reduction in both the size of the input data and in the computation time.

The main problem with supergather migration is that the crosstalk noise degrades the quality of the final image. Crosstalk noise is produced by migrating the ith shot gather with the imaging condition corresponding to the jth shot, where $j \neq i$. For example, if $S_i(\mathbf{x})$ is the modeled mono-frequency wavefield at $\mathbf{x}$ for an encoded point source at the ith shot position, then multisource modeling with one FD simulation yields $S(\mathbf{x}) = \sum_i S_i(\mathbf{x})$. Similarly, the back-propagated reflection wavefields in a supergather are given by $R(\mathbf{x}) = \sum_j R_j(\mathbf{x})$, where $R_j(\mathbf{x})$ represents the extrapolated wavefield at $\mathbf{x}$ associated for the jth source position. The mono-frequency supergather migration image is represented by

$$\tilde{\mathbf{m}}(\mathbf{x}) = S(\mathbf{x})R(\mathbf{x})^* = \sum_i S_i(\mathbf{x})R_i(\mathbf{x})^* + \sum_i \sum_{j \neq i} S_i(\mathbf{x})R_j(\mathbf{x})^*. \tag{2.1}$$

The single summation term represents the standard migration image where shot gathers are individually migrated and the images are stacked together, while the double summation represents the crosstalk noise that spoils the image quality. Here, the spectral product of any encoding function with its conjugate is selected so that it is equal to 1. The quality of $\tilde{\mathbf{m}}(\mathbf{x})$ can be improved by recomputing the supergather migration with different encoding functions and stacking this new image with the previously stacked image, but this increases the cost of the procedure.

Krebs et al. (2009) and Virieux and Operto (2009) empirically showed that multisource waveform inversion of phase-encoded supergathers[2] resulted in images with nearly the same quality as those of standard imaging, but with an order of magnitude less cost. An increase in efficiency was also obtained by Dai and Schuster (2009) and Dai et al. (2012) in applying least-squares reverse time migration to encoded supergathers. For the case of 2D waveform inversion, more than a 100-fold reduction in computation time was achieved in inverting 100-source supergathers. These results were empirically obtained, but did not rigorously reveal the theoretical relationship between the image quality and the encoding functions. Such an understanding could provide clues about how to optimally reduce crosstalk noise in the image from multisource migration or inversion.

In this chapter, formulas are derived that show how the quality of the migration image is related to the variance of the user-defined source statics, amplitude statics, and polarity statics for multisource imaging. Such formulas can be used to design optimal encoding

[2]Krebs et al. (2009) formed their n-fold supergathers by stacking n shot gathers, each with a random polarity statics for each source.

strategies for multisource imaging; for example, analysis reveals that the crosstalk term is exponentially reduced with increasing variance of the static shift and the square of the source frequency. We also provide formulas that relate the number of shot gathers in a superstack to the signal-to-noise ratio (SNR) of the final image. These insights suggest, e.g., that the variance of source statics should increase with coarser grids in multiscale inversion.

2.2 Phase-Encoded Migration of Supergathers

Assume a seismic experiment that consists of S shots, each shooting into the same number G of geophones. In the frequency domain the sth shot gather is denoted by the $G \times 1$ vector $\mathbf{d}_s$, where each element represents the frequency response at a different geophone. The relationship between the earth model and shot gather $\mathbf{d}_s$ is given by

$$\mathbf{d}_s = \mathbf{L}_s \mathbf{m}, \tag{2.2}$$

where $\mathbf{m}$ is the $N \times 1$ reflectivity model vector[3] and the $G \times N$ matrix $\mathbf{L}_s$ represents the prestack modeling operator for the sth shot gather.

The seismic experiment typically consists of S shots evenly distributed on the earth's surface, and each shot generates a new shot gather $\mathbf{d}_s$. This ensemble of shot gathers can be related to the earth model by the overdetermined system of equations

$$\begin{bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \\ \vdots \\ \mathbf{d}_S \end{bmatrix} = \begin{bmatrix} \mathbf{L}_1 \\ \mathbf{L}_2 \\ \vdots \\ \mathbf{L}_S \end{bmatrix} [\mathbf{m}] \quad \text{or} \quad \mathbf{d} = \mathbf{L}\mathbf{m}, \tag{2.3}$$

where $\mathbf{L}$ represents the $(S \cdot G) \times N$ overdetermined matrix of prestack forward modeling matrices, and $\mathbf{d}$ represents the $(S \cdot G) \times 1$ block vector of shot gathers. The standard migration procedure that approximates the solution to the above system of equations is known as prestack migration (Claerbout, 1992) and is represented in its simplest form by

$$\hat{\mathbf{m}} = \begin{bmatrix} \mathbf{L}_1^\dagger & & & \\ & \mathbf{L}_2^\dagger & & \\ & & \ddots & \\ & & & \mathbf{L}_3^\dagger \end{bmatrix} \begin{bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \\ \vdots \\ \mathbf{d}_3 \end{bmatrix} = \mathbf{L}^\dagger \mathbf{d} = \sum_{i=1}^{S} \mathbf{L}_i^\dagger \mathbf{d}_i, \tag{2.4}$$

where $\hat{\mathbf{m}}$ is the standard migration image, and $\mathbf{L}^\dagger$ represents the Hermitian conjugate of the modeling operator, otherwise known as the migration operator. The above equation

[3]The earth model is discretized into a regular grid of cells with an unknown reflectivity value m_i in the ith cell. There are N cells and the reflectivity model vector is defined as the $N \times 1$ vector $\mathbf{m}$.

says that each shot gather is separately migrated and the resulting migration images are summed together to give the standard prestack migration image.

The problem with 3D prestack migration is that it is computationally expensive for large models and large numbers of sources with high frequency content. To reduce this expense Romero et al. (2000) proposed that different shot gathers can be phase encoded, summed together to form a supergather, and then migrated all at once with a single finite-difference reverse time migration operation where the imaging condition is tuned to all of the different shot positions. This procedure is given by the following steps.

1. Phase encode the ith shot gather $\mathbf{d}_i$ by an *orthonormal* phase encoding function $\mathbf{N}_i$ such that $\langle \mathbf{N}_i^* \mathbf{d}_i^* \mathbf{N}_j \mathbf{d}_j \rangle = \mathbf{d}_i^* \mathbf{d}_j \delta_{ij}$, where $\langle \quad \rangle$ denotes ensemble averaging over the random phase encoding functions and δ_{ij} is the Kronecker delta function. For the discrete system of modeling equations, phase encoding shot gathers is equivalent to multiplying the ith block of modeling equations by the ith encoding function matrix $\mathbf{N}_i$ to form the encoded $(S \cdot G) \times G$ system of equations:

$$\begin{bmatrix} \mathbf{N}_1 & & & \\ & \mathbf{N}_2 & & \\ & & \vdots & \\ & & & \mathbf{N}_S \end{bmatrix} \begin{bmatrix} \mathbf{L}_1 \\ \mathbf{L}_2 \\ \vdots \\ \mathbf{L}_S \end{bmatrix} [\mathbf{m}] = \begin{bmatrix} \mathbf{N}_1 \mathbf{d}_1 \\ \mathbf{N}_2 \mathbf{d}_2 \\ \vdots \\ \mathbf{N}_S \mathbf{d}_S \end{bmatrix}, \tag{2.5}$$

or in more compact notation

$$[\mathbf{N}][\mathbf{L}]\mathbf{m} = [\mathbf{N}]\mathbf{d}. \tag{2.6}$$

Here, $\mathbf{N}_i$ is the $G \times G$ phase-encoding block matrix associated with the $G \times 1$ shot gather $\mathbf{d}_i$, and $[\mathbf{N}]$ represents the $S \cdot G \times G$ encoding matrix that contains all of the block matrices $\mathbf{N}_i$.

For a fixed frequency ω, the block-encoding matrix $\mathbf{N}_i$ can be reduced to the $G \times G$ identity matrix $\mathbf{I}$ multiplied by a scalar factor N_i:

$$\mathbf{N}_i = N_i \, \mathbf{I}. \tag{2.7}$$

An example of N_i is the phase shift $e^{i\omega\tau_i}$, where τ_i is a random time shift variable that time shifts the traces in the ith shot gather by the value τ_i. The important property of the phase-encoding matrices is that they are orthonormal under ensemble averaging; e.g., for the time shift encoding function we have

$$\langle \mathbf{N}_i \mathbf{N}_j^* \rangle = \langle e^{i\omega(\tau_j - \tau_i)} \rangle \, \mathbf{I} = \delta_{ij} \, \mathbf{I}, \tag{2.8}$$

where $\langle \quad \rangle$ represents ensemble averaging over the random variables τ_i and τ_j. An example of an encoded supergather is shown in Figure 2.1 with a) time-shifted shot

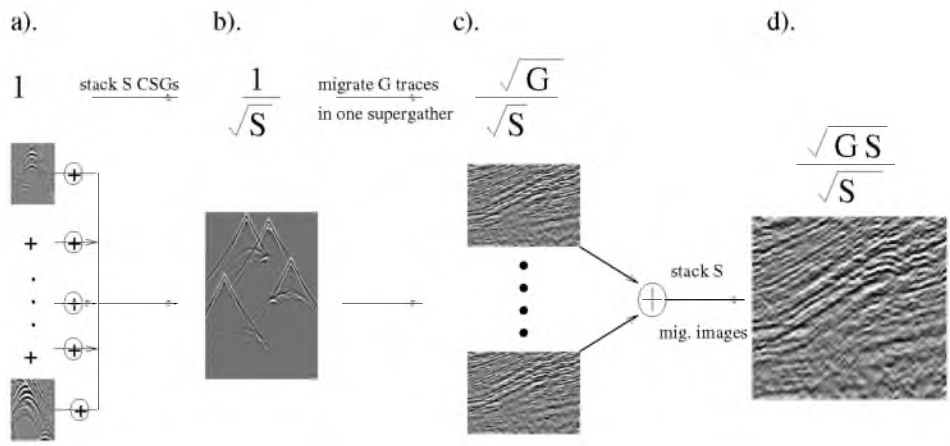

Figure 2.1. Blended shot migration: a) Time-shifted shot gathers, b) blended supergather created by blending and stacking S time-shifted shot gathers, c) migration images after migrating the supergather for each shot position and d) final image after summing S migration images. The final SNR enhancement is $\sqrt{G}$.

gathers and b) depicting the supergather created by blending (i.e., adding) the time-shifted shot gathers.

2. Summing the encoded shot gathers together can be achieved by multiplying the encoded $(S \cdot G) \times 1$ data vector in equation 2.6 by the $G \times (S \cdot G)$ blending matrix

$$\mathcal{I} = |\mathbf{I} \;\; \mathbf{I} \;\; \cdots \;\; \mathbf{I}| \tag{2.9}$$

to get

$$\overbrace{[\mathbf{N}_1\mathbf{L}_1 + \mathbf{N}_2\mathbf{L}_2 + \cdots \mathbf{N}_S\mathbf{L}_S]}^{\mathcal{L}=[\mathcal{I}][\mathbf{N}][\mathbf{L}]}\,\mathbf{m} = \overbrace{[\mathbf{N}_1\mathbf{d}_1 + \mathbf{N}_2\mathbf{d}_2 + \cdots \mathbf{N}_S\mathbf{d}_S]}^{\mathcal{D}=[\mathcal{I}][\mathbf{N}][\mathbf{d}]}; \tag{2.10}$$

or more compactly

$$\mathcal{L}\mathbf{m} = \mathcal{D}. \tag{2.11}$$

Here, the encoded $G \times N$ supergather modeling matrix $\mathcal{L}$ and the $G \times 1$ encoded supergather vector $\mathcal{D}$ have their row dimensions reduced by the multiplicative factor of S compared to the original system of equations 2.5.

3. The phase encoded migration of supergathers is obtained by applying the adjoint operator $\mathcal{L}^\dagger$ to the encoded supergather $\mathcal{D}$ to get

$$\begin{aligned}
\tilde{\mathbf{m}} &= \mathcal{L}^\dagger\mathcal{D} && (2.12)\\
&= \sum_{i=1}^{S}\mathbf{L}_i^\dagger\mathbf{N}_i^*\mathbf{N}_i\mathbf{d}_i + \sum_{j\neq i}^{S}\sum_{i=1}^{S}\mathbf{L}_j^\dagger\mathbf{N}_j^*\mathbf{N}_i\mathbf{d}_i \\
&= \overbrace{\sum_{i=1}^{S}\mathbf{L}_i^\dagger\mathbf{d}_i}^{\textit{standard migration } \hat{\mathbf{m}}} + \overbrace{\sum_{j\neq i}^{S}\sum_{i=1}^{S}\mathbf{L}_j^\dagger\mathbf{N}_j^*\mathbf{N}_i\mathbf{d}_i}^{\textit{crosstalk noise } \mathbf{c}} \\
&= \hat{\mathbf{m}} + \mathbf{c}, && (2.13)
\end{aligned}$$

which says that the phase-encoded migration image $\tilde{\mathbf{m}}$ consists of the standard migration image $\hat{\mathbf{m}}$ in equation 2.4 and the crosstalk term $\mathbf{c}$. In this case it is assumed that $\mathbf{N}_i^*\mathbf{N}_i = \mathbf{I}$ for all integer values of i.

If the norm of the crosstalk vector $\mathbf{c}$ is small then the phase-encoded image $\tilde{\mathbf{m}}$ in equation 2.13 (obtained from an S stack supergather) closely approximates the standard prestack migration image $\hat{\mathbf{m}}$. Thus, encoded migration of this supergather might result in an S-fold reduction in computation time compared to separately migrating each shot gather.

2.3 Iterative Phase-Encoded Migration of Supergathers

The crosstalk vector $\mathbf{c}$ in equation 2.13 is typically not small in magnitude for large numbers of encoded shot gathers in the supergather, and so the resulting migration image contains unacceptable noise (Romero et al., 2000). To decrease the level of this noise, iterative migrations can be performed and two such methods are examined: iterative stacking of encoded migration images and iterative least-squares migration of encoded supergathers.

2.3.1 Iterative Stacking of Encoded Migration Images

Several encoded migration images can be computed using equation 2.13, but each one dynamically uses a new encoding function[4] to form the input supergather. These images can be added (i.e., stacked) to the previous one. If this process is iterated, then the $k+1$ stacked migration image $\tilde{\mathbf{m}}^{(k+1)}$ obtained by stacking k realizations of equation 2.13 is given by:

$$\tilde{\mathbf{m}}^{(k+1)} = \tilde{\mathbf{m}}^{(k)} + \hat{\mathbf{m}} + \mathbf{c}^{(k+1)}, \tag{2.14}$$

which, after backward substitution, can be expressed as

$$\begin{aligned} \tilde{\mathbf{m}}^{(k+1)} &= \overbrace{(k+1)\hat{\mathbf{m}}}^{image\ signal} + \overbrace{\sum_{k'=1}^{k+1} \mathbf{c}^{(k')}}^{crosstalk\ noise} \\ &= (k+1)\hat{\mathbf{m}} + \sum_{j\neq i}^{S}\sum_{i=1}^{S} \mathbf{L}_j^{\dagger} \,[\sum_{k'=1}^{k+1} \mathbf{N}_j^{*(k')}\mathbf{N}_i^{(k')}]\, \mathbf{d}_i. \end{aligned} \tag{2.15}$$

The block matrix $\mathcal{N}_{ij} = \sum_{k'=1}^{k+1} \mathbf{N}_j^{*(k')}\mathbf{N}_i^{(k')}$ approximates the ensemble averaging of the covariance matrix associated with the encoding functions. Since the phase encoding function $\mathbf{N}_i$ is uncorrelated with $\mathbf{N}_j$ for $j \neq i$, then $\mathcal{N}_{ij}$ grows as $\sqrt{k+1}$ compared to the migration signal $(k+1)\hat{\mathbf{m}}$ growth of $k+1$. Hence, the SNR of the image associated with equation 2.15 should grow as $\sqrt{k}$.

2.3.2 Iterative Least-squares Migration

The overdetermined system of equations $\mathbf{Lm} = \mathbf{d}$ is often inconsistent and so an iterative least-squares solution can be computed to form what is known as the least-squares migration

[4]The crosstalk term at the k iteration is computed by using the k realization of the random variable τ_i.

(Nemeth et al., 1999; Duquet et al., 2000). The LSM image for phase-encoded supergathers is obtained by first forming a data misfit function

$$\epsilon \;=\; \frac{1}{2}(\mathcal{L}\mathbf{m}-\mathcal{D})^{\dagger}(\mathcal{L}\mathbf{m}-\mathcal{D}), \tag{2.16}$$

so that the iterative steepest descent solution at the $k+1$ iteration is given by

$$\begin{aligned}\mathbf{m}^{(k+1)} &= \mathbf{m}^{(k)} - \alpha^{(k)}\mathcal{L}^{(k)\,\dagger}(\mathcal{L}^{(k)}\mathbf{m}^{(k)} - \mathcal{D}^{(k)}) \\ &= \mathbf{m}^{(k)} - \alpha^{(k)} \overbrace{\sum_{i=1}^{S}\mathbf{L}_i^{\dagger}\delta\mathbf{d}_i^{(k)}}^{standard.\ LSM} - \alpha^{(k)} \overbrace{\sum_{j\neq i}^{S}\sum_{i=1}^{S}\mathbf{L}_j^{\dagger}\mathbf{N}_j^{*(k)}\mathbf{N}_i^{(k)}\delta\mathbf{d}_i^{(k)}}^{crosstalk\ noise\ \mathbf{c}}.\end{aligned} \tag{2.17}$$

Here, the data residual at the kth iteration for the ith shot gather is defined as

$$\delta\mathbf{d}_i^{(k)} \;=\; \mathbf{L}_i\mathbf{m}^{(k)} - \mathbf{d}_i \;=\; \mathbf{L}_i(\mathbf{m}^{(k)} - \mathbf{m}), \tag{2.18}$$

and $\alpha^{(k)}$ is the scalar step length. Without losing generality but gaining simplicity, we assume that the step length $\alpha^{(k)}$ is constant for all iterations such that $0 < \alpha < 2/\lambda_{max}$, where λ_{max} is the maximum eigenvalue of the well-conditioned symmetric positive definite (SPD) matrix $\mathcal{L}^{\dagger}\mathcal{L}$. This procedure is sometimes known as the *fixed step size steepest descent method* and is guaranteed to converge for SPD $\mathcal{L}^{\dagger}\mathcal{L}$ (Burden and Faires, 2001). Also, we assume that the crosstalk term is small enough so that phase-encoded steepest descent solution converges to a model close to the actual standard LSM model[5] for a reasonable number of iterations.

Substituting equation 2.18 into equation 2.17, assuming a fixed step length α that satisfies the above convergence condition, and employing backward substitution yields

$$\begin{aligned}\mathbf{m}^{(k+1)} &= \mathbf{m}^{(0)} - \alpha\sum_{i=1}^{S}\mathbf{L}_i^{\dagger}\mathbf{L}_i \overbrace{\{\sum_{k'=0}^{k}[\mathbf{m}^{(k')} - \mathbf{m}]\}}^{\mathbf{a}} \\ &\quad -\alpha\sum_{j\neq i}^{S}\sum_{i=1}^{S}\mathbf{L}_j^{\dagger}\mathbf{L}_i \overbrace{\{\sum_{k'=0}^{k}c_{ij}^{(k')}[\mathbf{m}^{(k')} - \mathbf{m}]\}}^{\mathbf{b}}.\end{aligned} \tag{2.19}$$

Here, $\mathbf{a}$ is the sum of model residuals; $\mathbf{b}$ is the sum of randomized model residuals; $c_{ij}^{(k')}\mathbf{I} = \mathbf{N}_j^{*(k')}\mathbf{N}_i^{(k')}$, and $c_{ij}^{(k')} = e^{i\omega(\tau_i^{(1k')} - \tau_j^{(k')})}$ for random time shift encoding.

---

[5]Rigorous error bounds for $\delta\mathbf{x}$ and the perturbed system of equations $[\mathbf{A}+\delta\mathbf{A}](\mathbf{x}+\delta\mathbf{x}) = \mathbf{b}+\delta\mathbf{b}$ are given in Burden and Faires (2001). Correspondingly, the crosstalk term in equation 2.17 can be related to $\delta\mathbf{A}$ while the standard migration term can be related to $\mathbf{A}$. The key point they make is that for well-conditioned $\mathbf{A}$ and a small crosstalk term $\delta\mathbf{A}$, $\delta\mathbf{x}+\mathbf{x}$ is close to $\mathbf{x}$.

In equation 2.19, the bracketed *sum of model residuals* vector $\mathbf{a}$ should be compared to the last term $\mathbf{b}$ in brackets, which is a randomized sum of model residual vectors. If the sequence of model residuals $\mathbf{m}^{(k')} - \mathbf{m}$ are nearly parallel[6] to one another and are approximately equal to $\delta\mathbf{m}$, then $\mathbf{a}$ and $\mathbf{b}$ in equation 2.19 can be approximated as

$$\mathbf{a} \approx (k+1)\delta\mathbf{m}; \quad \mathbf{b} \approx \delta\mathbf{m} \sum_{k'=0}^{k} c_{ij}^{(k')}. \tag{2.20}$$

Similar to the crosstalk noise in equation 2.15, this noise vector $\mathbf{b}$ decreases compared to the signal vector $\mathbf{a}$ as the iteration index k increases. For this example, the iterations in the steepest descent method are a proxy[7] for ensemble averaging seen in equation 2.15.

The relative benefit of phase-encoded iterative LSM over iterative stacking of encoded migration images is that, for an accurate velocity model and a well-conditioned and mostly consistent[8] system of equations, it produces images with higher spatial resolution. Empirical results (Dai et al., 2011; Wang and Schuster, 2012) also suggest that it converges faster than iterative stacking.

The practical implementation of iterative LSM applied to encoded supergathers is the following.

1. Choose a starting vector $\mathbf{m}^{(k)}$ for $k = 0$.
2. Select the kth realization for the values of the random variables $[\tau_1,\ \tau_2,\ \cdots \tau_S]^{(k)}$, and compute the encoded supergather data $\mathcal{D}^{(k)}$.
3. Compute the modeled supergather $\mathcal{L}\mathbf{m}^{(k)}$ with S encoded sources.
4. Use the results of the supergather forward modeling to subtract from the phase encoded data $\mathcal{D}^{(k)}$ to get the residual $\mathbf{r}^{(k)} = \mathcal{L}\mathbf{m}^{(k)} - \mathcal{D}^{(k)}$ in equation 2.17.
5. Migrate the residual using $\mathcal{L}^{\dagger}\mathbf{r}^{(k)}$ to update the model with equation 2.17.
6. Set $k \rightarrow k+1$ and repeat steps 2-5 until acceptable convergence.

---

[6]This special case is an ideal gedanken example used to guide, not prove, our intuitive understanding of convergence. Strict convergence criteria for stochastic gradients and approximation are discussed in Kushner and Yin (2003), where at each iteration the gradient is averaged over different realizations of the random encoding functions. This is somewhat similar to the method suggested by equation 2.19.

[7]This is similar to the ergodic assumption that the ensemble average of the random variable $x(t)$ at a specific value of t, where $x(t)$ is a stationary white noise series, can be approximated by a weighted summation of $x(t)$ over the time index.

[8]Inconsistency becomes worse with increasing error in the migration velocity model.

2.4 Signal-to-Noise Ratio vs Number of Blended Shot Gathers

2.4.1 Theory

Equation 2.15 can be used to deduce formulas for SNR enhancement[9] of the migration image as a function of the number I of stacking iterations and the number S of encoded shot gathers in a supergather. For this analysis, we will assume the ideal condition that each trace is contaminated[10] with additive white noise and also contains a band-limited signal with amplitude 1. There are G traces/shot gather, S time-shifted shot gathers/supergather, and all of the G evenly spaced geophones are fixed in location and record G traces/shot.

Figure 2.1a-b illustrates that stacking S time-shifted shot gathers together[11], will degrade the SNR ratio of a supergather migration image by the factor proportional to $1/\sqrt{S}$; Figure 2.1c demonstrates that migrating a supergather of G traces for one assumed shot position will enhance the migration image trace to give a SNR ratio proportional to $\sqrt{G}/\sqrt{S}$; and d) illustrates that stacking S migration images together gives a SNR enhancement of $\sqrt{GS}/\sqrt{S} = \sqrt{G}$. To further enhance the SNR ratio, I stacks of migration images, each with different realizations of different encoding functions, will give the final SNR enhancement proportional to

$$(SNR)^{super} = \sqrt{GI}. \tag{2.21}$$

In comparison, the conventional prestack migration of S shot gathers (each with G traces) gives a SNR enhancement proportional to

$$(SNR)^{stand.} = \sqrt{GS}. \tag{2.22}$$

This assumes that the signal in all of the traces is of the same amplitude and that the geometrical spreading effects have been corrected. Under the ideal conditions stated, phase-encoded migration (equation 2.21) requires the same number of iterations as the number of shot gathers S in a supergather in order to achieve the same SNR level seen in standard prestack migration (equation 2.22).

An important observation is that $(SNR)^{super} = \sqrt{GI}$ is independent of the number of shots that comprise the supergather. Therefore, the original supergather can be subdivided

---

[9]SNR is defined as $S/N = \|\mathbf{m}_{\text{ref}}\|/\|\tilde{\mathbf{m}}^{(k+1)} - \mathbf{m}_{\text{ref}}\|$, where $\tilde{\mathbf{m}}^{(k+1)}$ is the $(k+1)$-fold stacked image, and $\mathbf{m}_{\text{ref}}$ is the reference migration image (see Section 2.4.2 for details).

[10]We assume a zero-mean white noise series governed by a normal probability density function with a standard deviation of 1.

[11]The time shift is assumed to have a standard deviation greater than the source period.

into Q sub-supergathers[12], each one created by blending a unique set of S/Q encoded shot gathers. Iteratively migrating each of these Q sub-supergathers and stacking the images together will lead to an enhanced

$$SNR^{sub-super} = \sqrt{QGI}, \tag{2.23}$$

but at the expense of Q extra supergather migrations. Results shown in Dai et al. (2011) and Boonyasiriwat and Schuster (2010) are consistent with this prediction.

2.4.2 Numerical Tests

To show that the above formula roughly predicts the SNR as a function of the number of encoded image stacks, a 320-source phase-encoded supergather is generated by a Born modeling code for the SEG/EAGE salt model and iteratively migrated and stacked according to equation 2.15. The spacing between the evenly spaced sources is 18.3 m, which is about 61% of a wavelength for the 50 Hz Ricker wavelet; the standard deviation of the source static is 0.4 second, which is about 20 times of the source's dominant period. At each iteration of equation 2.15, a new realization of the encoding functions is used to time shift each shot gather, followed by blending to form the supergather and its migration. After each iteration, the stacked migration image $\mathbf{m}^{(k)}$ is subtracted from the reference migration image $\mathbf{m}_{\text{ref}}$ to give the noise $\mathbf{m}^{(k)} - \mathbf{m}_{\text{ref}}$, which is then used to calculate SNR=$\|\mathbf{m}_{\text{ref}}\|/\|\mathbf{m}^{(k)} - \mathbf{m}_{\text{ref}}\|$, where $\mathbf{m}_{\text{ref}} = \lim_{p\to\infty} \mathbf{m}^{(p)}$. Note that because there exists a residual discrepancy between $\lim_{p\to\infty} \mathbf{m}^{(p)}$ and the standard migration image $\tilde{\mathbf{m}}$, $\mathbf{m}^{(p)}$ for a large value of p is chosen as $\mathbf{m}_{\text{ref}}$ in order to single out the effect of the number of stackings on SNR. Figure 2.2a shows the standard migration image, Figure 2.2b shows the one supergather migration image and Figure 2.2c-h shows the multisource migration image after 30, 60, 90, 120, 150 and 180 iterative stackings. The crosstalk noise decreases with more iterations as describes by equation 2.21. In Figure 2.3, the solid line shows the measured SNR and largely agrees with the dashed line predicted by equation 2.21, where the $16,291$ stacked image is used as $\mathbf{m}_{\text{ref}}$. Note the wiggles in the solid curve confirm the prediction that the estimated SNR has nonzero variance for reasonably large value of $\sigma\omega$ (see Appendix B in Schuster et al. (2011)).

[12]Here, S/Q is assumed to be an integer number.

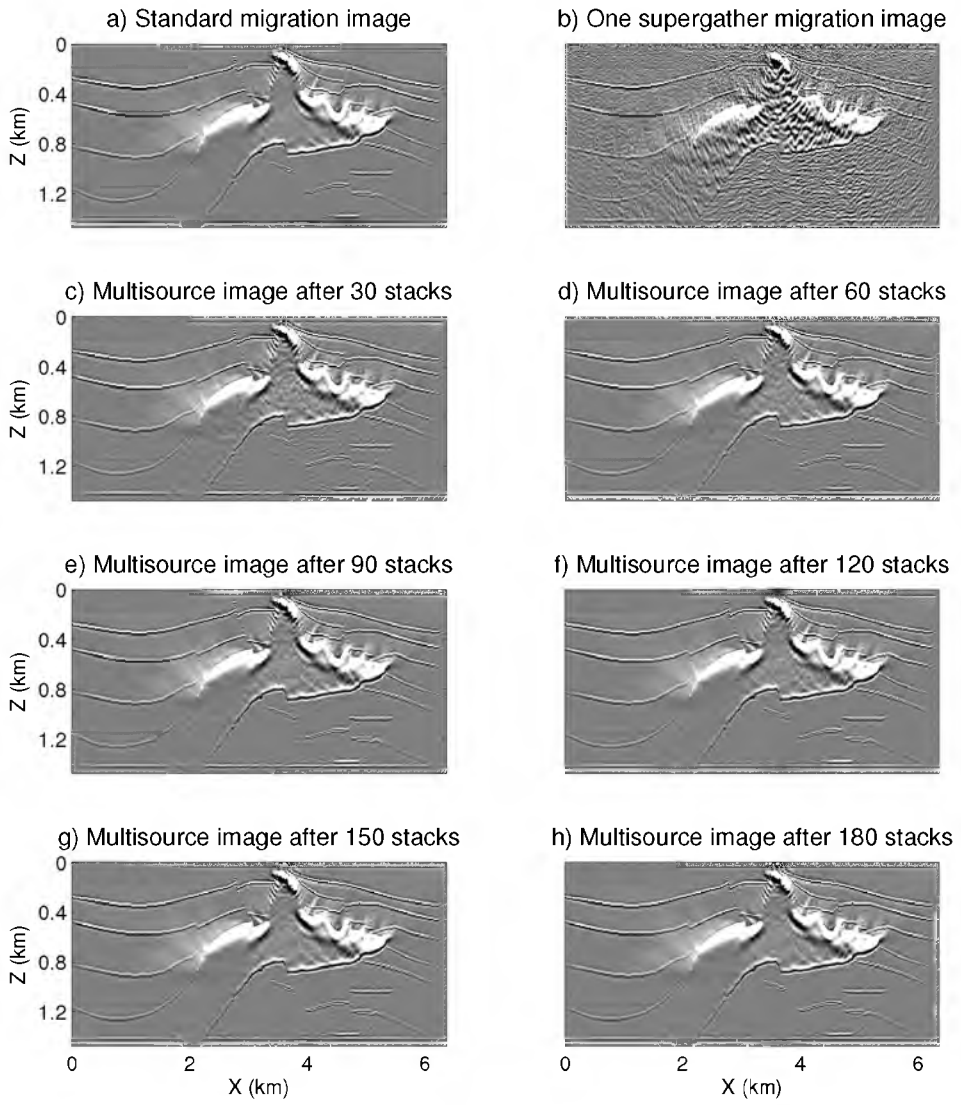

Figure 2.2. Phase-encoded iterative migration images for the SEG salt model: a) standard migration, b) one supergather migration and c-h) iterative stacking images after 30, 60, 90, 120, 150 and 180 iterations. Here the standard deviation of the source statics is 0.05 second.

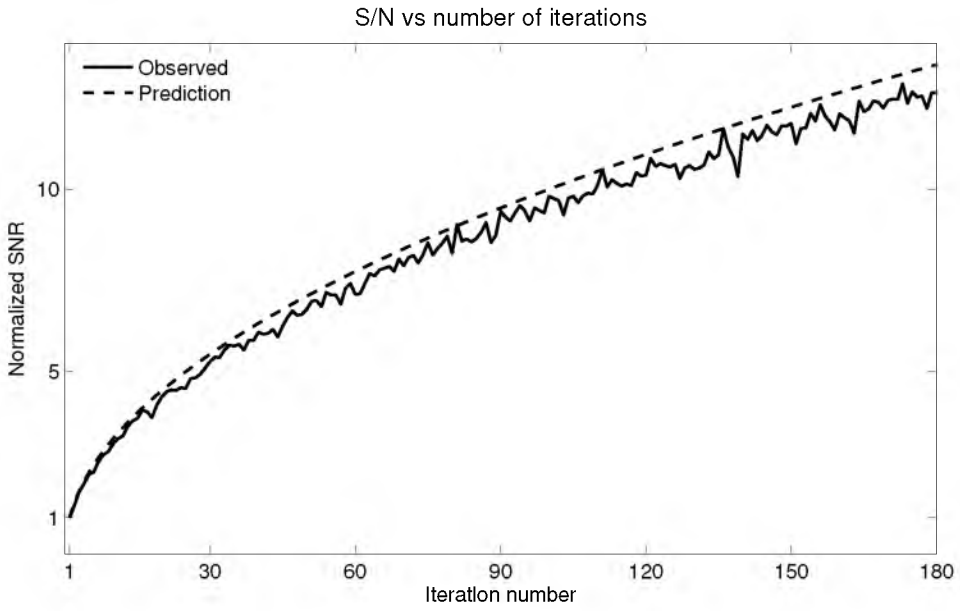

Figure 2.3. The solid (dotted) line indicates the measured SNR for phase-encoded iterative stacking migration, where the input data set consists of a 320-shot supergather associated with the SEG/EAGE salt model; the dashed line indicates the predicted relation from equation 2.21. Both curves have been normalized to the value 1 for the first iterate.

2.5 Ensemble Averaging and Crosstalk Noise

Under the white-noise assumptions described in the previous section, the averaged crosstalk term in equation 2.15 becomes small with increasing number k of stacks, so the phase-encoded image approaches that of the standard migration image $m(\mathbf{x})$. This can be shown by taking the ensemble average of equation 2.15:

$$\langle \tilde{\mathbf{m}} \rangle = \overbrace{\sum_{i=1}^{S} \mathbf{L}_i^\dagger \mathbf{d}_i}^{stand.migration} + \overbrace{\sum_{j \neq i}^{S} \sum_{i=1}^{S} \mathbf{L}_j^\dagger \mathbf{d}_i \underbrace{\langle N_j^* N_i \rangle}_{R}}^{crosstalk\ noise}, \tag{2.24}$$

where $R = \langle N_j^* N_i \rangle$ is defined as the *crosstalk damping coefficient*[13]. Equation 2.24 follows because equation 2.7 is henceforth assumed and thus $\mathbf{N}_j^* \mathbf{N}_i = N_j^* N_i \mathbf{I}$. In this section, I derive the formula for R, which determines the mean value of crosstalk noise is derived, for time-shift statics and for polarity statics.

As a background, $\Phi(\omega) \stackrel{\text{def}}{=} \langle e^{i\omega\tau} \rangle_\tau$ is known as the *characteristic function* of a random variable τ, and if τ is normally distributed with zero-mean and σ^2-variance, denoted in shorthand as $\tau \sim \mathcal{N}(0, \sigma^2)$, its characteristic function is given by [see, e.g., eq. (8-57) in Papoulis and Pillai (1991)]:

$$\Phi(\omega) = e^{-\omega^2 \sigma^2 / 2} \tag{2.25}$$

Recall also the fact that if two random variables u and v are mutually independent, then

$$\langle f(u) g(v) \rangle = \langle f(u) \rangle \langle g(v) \rangle, \tag{2.26}$$

where $f(\cdot)$ and $g(\cdot)$ are two arbitrary functions that admit well-defined expectation value.

2.5.1 Source Time Statics

2.5.1.1 Theory

$N_s = e^{i\omega\tau_s}$. Assume that $\tau_s \sim \mathcal{N}(0, \sigma^2)$ represents a time shift applied to the sth source, and that the random variables τ_s, $\forall s = 1, \ldots, S$ are independent and identically distributed (i.i.d.). Note that $\langle N_s \rangle$ then happens to be the characteristic function $\Phi(\omega)$ given in equation 2.25. Note also

$$N_s^* N_s = 1 \tag{2.27}$$

---

[13] The subscripts $_{ji}$ of R are omitted for notational brevity where no ambiguity exists.

since N_s is of pure phase. The expected cross-correlation of the phase-encoding function can be readily calculated as

$$\begin{aligned} R &= \langle N_s^* \rangle \langle N_{s'} \rangle, \text{ if } s \neq s', \\ &= |\langle N_s \rangle|^2 \\ &= e^{-\omega^2 \sigma^2}, \end{aligned} \tag{2.28}$$

where the first equality follows from equation 2.26, and the last step follows from equation 2.25. The foregoing results are summarized as

$$R = \begin{cases} 1 & \text{if } s = s', \\ e^{-\omega^2 \sigma^2} & \text{if } s \neq s'. \end{cases} \tag{2.29}$$

This equation resembles a Kronecker-δ function, i.e., $R \approx \delta_{ss'}$, if $\sigma \gg \omega^{-1}$. Equation 2.24 in conjunction with equation 2.29 thus says that the crosstalk noise can be reduced if the standard deviation σ of the source static shift is much greater than the period $2\pi/\omega$. Therefore, the variance of the user-specified source statics should increase with coarser grids in a multiscale inversion strategy.

In practice, ensemble averaging is not explicitly performed with supergather migration because this would be too costly. A poor man's proxy for ensemble averaging is the iterative stacking or iterative least-squares migration described in the previous sections, or the iterative updates characteristic of iterative waveform inversion (Tarantola (1984); Mora (1987); Krebs et al. (2009)).

2.5.1.2 Numerical Tests

Numerical experiments are used to test the validity of equation 2.29 for encoded supergathers associated with a single scatterer model as well as the SEG/EAGE salt model. A Born modeling solution to the 2D wave equation is used to compute the synthetic data with both the sources and geophones on the surface. The scatterer is at a depth of 0.5 km in a homogeneous model with the P-wave velocity of 2000 m/s, and the dominant frequency of the narrow-band Ricker wavelet is 5 Hz; for the salt model the dominant frequency of the narrow-band Ricker source wavelet[14] is 50 Hz. Geophones are evenly distributed on the top surface of each model, with 256 and 320 traces per shot gather for the point scatterer model and the salt model, respectively.

The supergathers consist of 256 and 320 blended shot gathers for the point scatterer model and the salt model, respectively. Each supergather uses a new realization of the source

[14]Narrow band sources are used because equation 2.29 is only valid for a single frequency source.

statics and is migrated, and the resulting migration images are stacked together. Figure 2.4 shows the standard migration and iterative stacking images with source statics of 0.01 s, 0.03 s, 0.05 s, 0.08 s and 0.1 s for the single scatterer model. As expected, the migration images become more focused as σ increases because greater time shifts in neighbouring shot gathers reduces their similarity as well as the crosstalk strength in the migration image. This statement is further validated in Figure 2.5, which plots both the theoretical (i.e., equation 2.24) and numerically calculated crosstalk strength R[15] as a function of σ for the source time statics. There is very close agreement between the theoretical and numerically calculated curves.

The dynamically encoded supergathers associated with the salt model are also migrated and iteratively stacked to give the images shown in Figure 2.6. As expected, the images become less noisy with an increase in the number of stacks and (as shown in Figure 2.7) the standard deviation of the source statics. It is expected that the crosstalk noise in the salt image should be much greater than that for the single scatterer because more scatterers will increase the $\mathbf{L}_i\mathbf{d}_j$ contributions in the crosstalk. This observation might partly explain the curves in Figure 2.7, which show that the absolute strength of the crosstalk noise with respect to σ is not well predicted by equation 2.24. Better agreement can be reached with a greater number of stacks, but limited computational resources prevented a thorough testing of this claim. Moreover, the theoretical prediction from equation 2.24 is for a single frequency at 50 Hz, while the simulations used a narrow-band 50 Hz source with a bandwidth of 26.25 Hz.

The R computed from the salt-model simulations is still higher than the prediction, although there is a trend of increasing agreement as the number of stacks grows. Despite this discrepancy, the multisource migration image after 500 stackings is visually indistinguishable from the single-source counterpart, as evident in Figure 2.6. Note the excessive ringing in the images is due to the narrow band nature of the source wavelet.

2.5.2 Polarity Statics

2.5.2.1 Theory

$N_s = p_s = \pm 1$. Here, the polarity p_s is a binary random variable that is i.i.d., $\forall\, s = 1, \ldots, S$, and p_s takes the values of -1 or 1 with probability 50% for each. It can be readily verified that

[15] What is plotted in Figure 2.5 and 2.7 is normalized R, defined as $\hat{R} = \|\mathbf{m}_\sigma^{(k)} - \tilde{\mathbf{m}}\| / \|\mathbf{m}_{\sigma_0}^{(k)} - \tilde{\mathbf{m}}\|$, where $\sigma_0 = 0.01$s and $\tilde{\mathbf{m}}$ is the standard migration image.

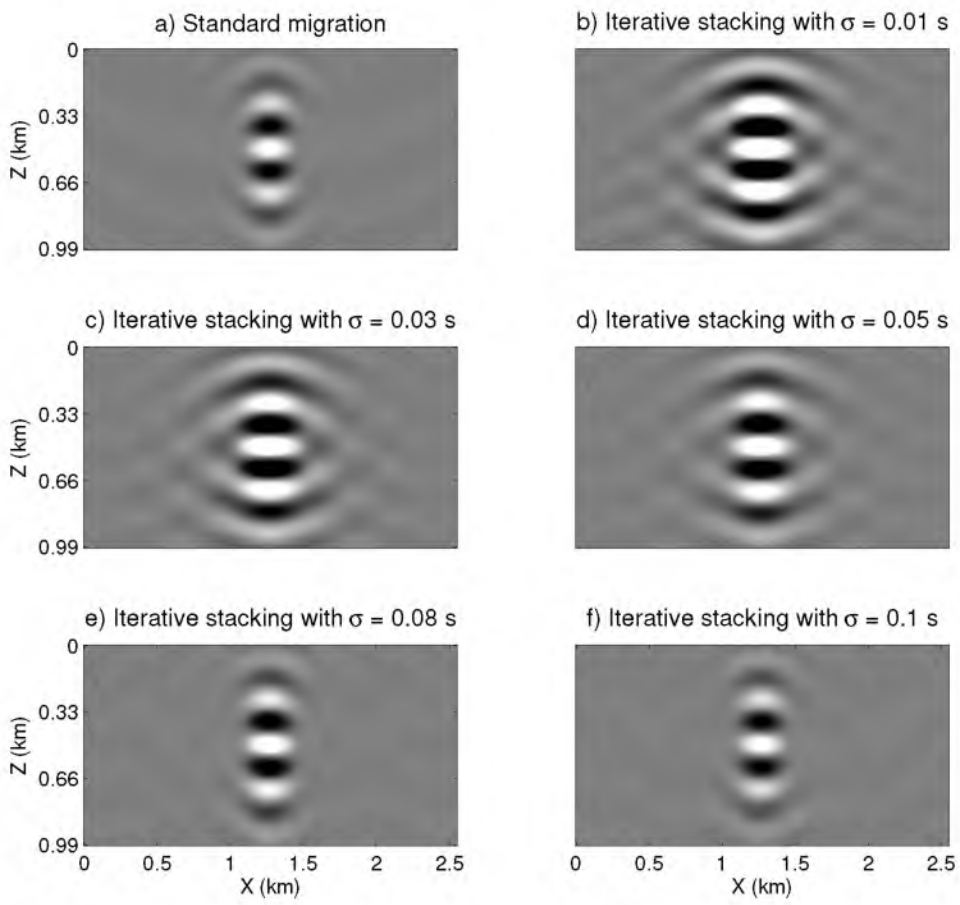

Figure 2.4. Iterative stacking migration images for data associated with the point scatterer model and the 5 Hz narrow band source wavelet: a) standard migration, b) - f) show the iterative stacking images for the standard deviation of source static of 0.01 s, 0.03 s, 0.05 s, 0.08 s and 0.1 s. Only 100 stacks were needed to compute these images.

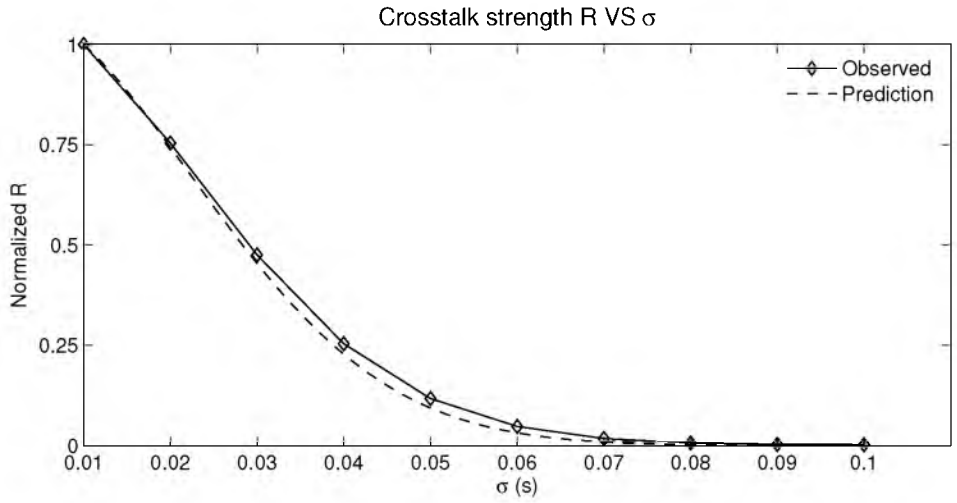

Figure 2.5. The solid line with diamonds indicates the measured normalized crosstalk damping coefficients for phase-encoded iterative stacking migration, where the input data set consists of a 256-shot supergather associated for a point scatterer model with a homogeneous background velocity of 2000 m/s; the dashed line indicates the predicted relation from equation 2.21, and the input source wavelet is band limited with a dominant frequency of 5 Hz. The observed curve is nearly the same for a range of τ_s standard deviations between 0.01 s and 0.1 s.

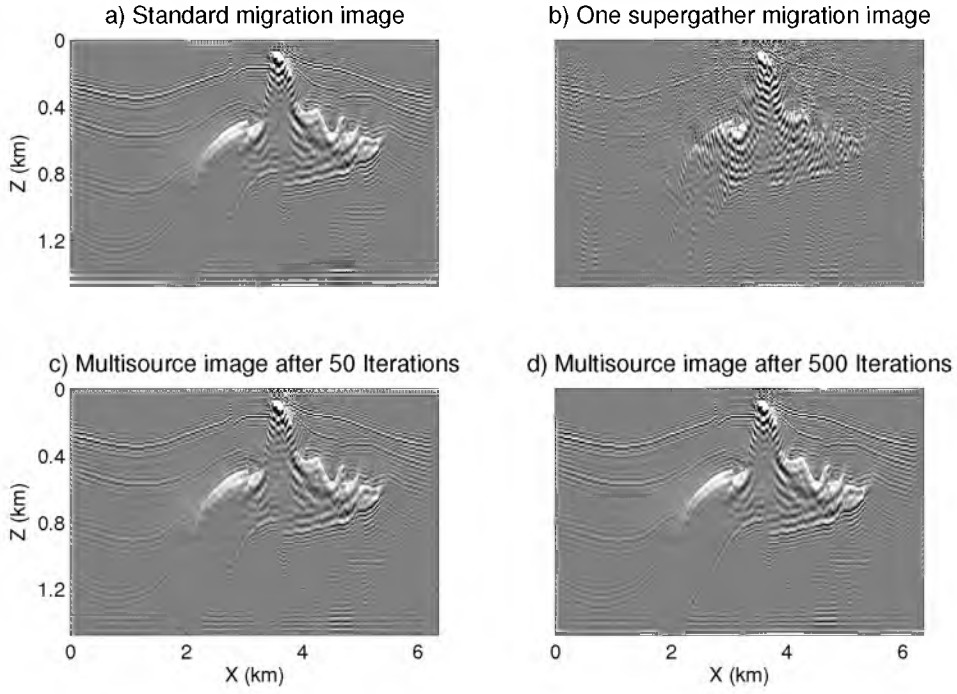

Figure 2.6. Iterative stacking migration images for the SEG salt model, where the narrow band source wavelet is peaked at 50 Hz: a) standard migration, b), c), and d), iterative stacking images after 1, 50, 500 iterations. Here, the standard deviation of the source statics is 0.05 second.

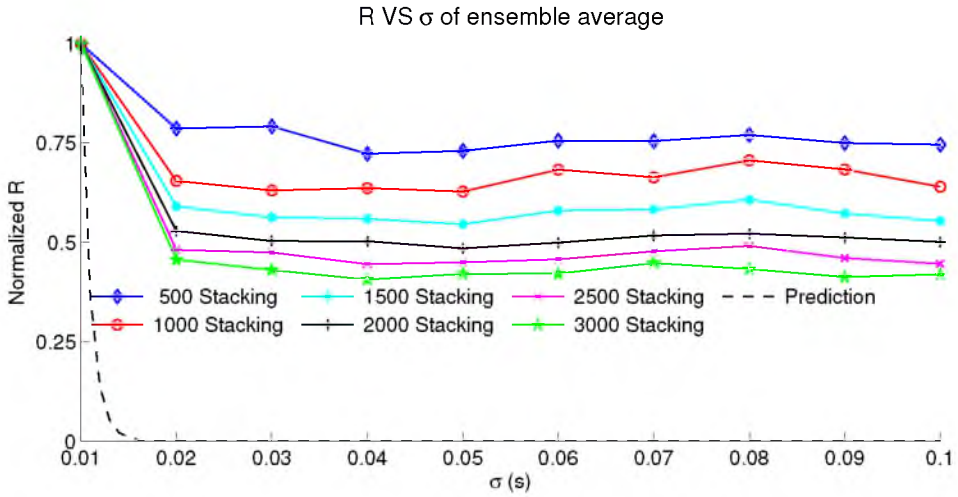

Figure 2.7. The solid lines with different colors indicate the measured normalized crosstalk damping coefficients for phase-encoded iterative stacking migration after 500, 1000, 1500, 2000 2500, and 3000 stacks, where the input data consists of a 320-shot supergather associated with the SEG salt model. The dashed line indicates the predicted relation from equation 2.24, where the input narrow band source wavelet is peaked at 50 Hz.

$$R = \langle p_s p_{s'} \rangle = \delta_{ss'}, \tag{2.30}$$

which shows that the crosstalk (i.e., when $s \neq s'$) noise is of 0-mean. Compared to the corresponding result of time statics, expressed in equation 2.24, equation 2.30 is an exact Kronecker-δ function, rather than an approximation as in equation 2.24. This means that the performance of polarity encoding should be superior to that of time statics for low frequencies.

That the crosstalk noise has 0-mean is also true if the polarity statics act in conjunction with other encoding statics as long as they are mutually independent. Namely, if $N_s = p_s h(\tau_s)$, where $h(\tau_s)$ is the other static and p_s is independent of τ_s, then it follows from equation 2.26 that the crosstalk damping coefficient for shot gather encoded by both polarity and $h(\tau_s)$ is given by

$$\begin{aligned} R &= \langle p_s p_{s'} \rangle \langle h(\tau_s)^* h(\tau_{s'}) \rangle, \\ &= \langle |h(\tau_s)|^2 \rangle \delta_{ss'}. \end{aligned} \tag{2.31}$$

This opens up new possibilities for combined encoders such as applying both polarity and time statics to each shot gather. Therein the energy (or variance, in the case of 0-mean) of the resulting crosstalk noise can be used to quantify the effectiveness of candidate encoders. This is because a relationship exists between the variance of the crosstalk noise and the required number of stacks to meet a desired criterion (e.g., SNR) – the smaller the former is, the smaller the latter will be. Schuster et al. (2011) showed that the variance of the combined encoder of polarity and time statics is lower-bounded by 50% of the variance of polarity encoding alone. This means that in the ideal case the former encoding strategy can be twice as good as the latter one.

2.5.2.2 Numerical Tests

The merit of encoding by combined polarity and time statics is substantiated by numerical tests on the salt model. Figure 2.8 shows that the combined encoder of polarity and time statics produces a better migration image than computed with either polarity or time statics alone. For example, the residual error associated with the shallow reflector is significantly reduced in panel c2) as compared to panel b2).

The SNR[16] curves, as shown in Figure 2.9, validate the theoretical prediction (Schuster et al., 2011), that combining both polarity and statics is the superior encoding strategy. In

[16]Here, SNR is defined as $\|\hat{\mathbf{m}}\|/\|\boldsymbol{\delta}\|$, where $\hat{\mathbf{m}}$ and $\boldsymbol{\delta}$ are defined in the caption of Figure 2.8.

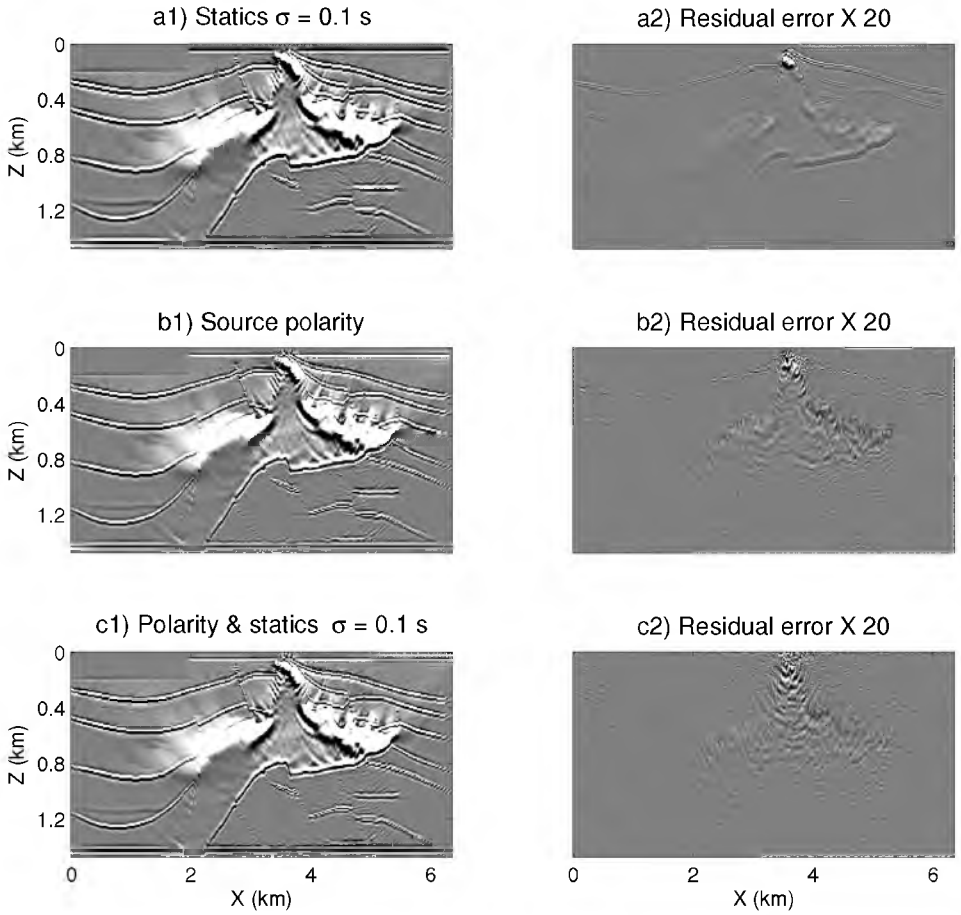

Figure 2.8. Comparison of iterative migration images $\mathbf{m_{enc}}$ with: a1) time statics encoding ($\sigma = 0.1$ s), b1) source polarity encoding and c1) the combination of both. b1), b2) and b3) show the residual error $\boldsymbol{\delta}$, i.e., the difference between the standard single-source migration $\hat{\mathbf{m}}$ and $\mathbf{m_{enc}}$, respectively. All iterative migration images are after 3000 stacks. The residual error is scaled by a factor of 20, for ease of perception, in b2) and c2). The input data set consists of a blended 320-shot supergather associated with the SEG salt model, and the input source wavelet is a Ricker wavelet with peak frequency at 50 Hz.

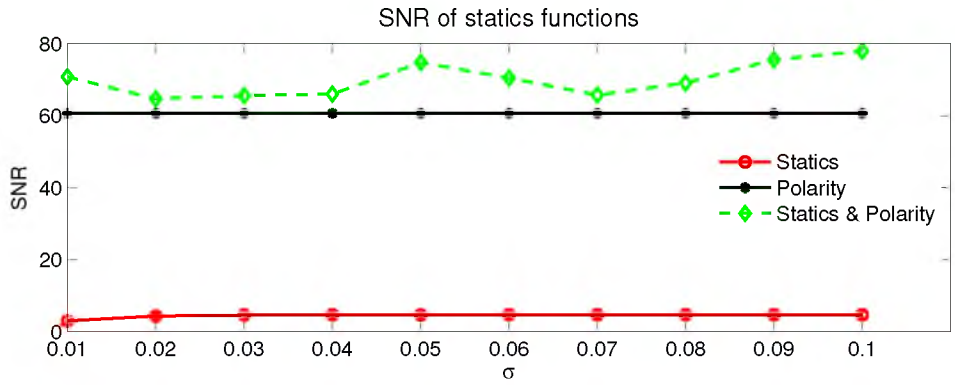

Figure 2.9. SNR vs the standard deviation of time statics, obtained from iterative migration after 3000 stacks, using encoding techniques such as: (bottom curve, solid, with diamonds) time statics, (middle line, solid) polarity, and (top curve, dashed, with diamonds) combined time statics and polarity. The input data set consists of a blended 320-shot supergather associated with the SEG salt model, and the input source wavelet is a Ricker wavelet with peak frequency at 50 Hz.

this figure, several features are noteworthy. First, the bottom curve (associated with time statics) is significantly lower than the other two curves. The reason is that the source wavelet has a low frequency content, which resists the crosstalk damping factor R, since it follows from equation 2.28 that $\lim_{\omega \to 0} R = 1$. Second, the solid line in the middle is constant with respect to σ, because the polarity encoding is governed by random variables independent of those governing time statics. Third, the apex of the top curve reads $\max \mathrm{SNR}_{pt} \simeq 78$, where the subscript $_{pt}$ denotes combined polarity and time statics. Let SNR_p be the SNR produced with polarity encoding, which is about 60.5 in Figure 2.9. Therefore we have $\max \mathrm{SNR}_{pt}^2 / \mathrm{SNR}_p^2 \simeq (78/60.5)^2 = 1.66 < 2$, in line with inequality of B30 discussed in Schuster et al. (2011).

2.6 Static, Hybrid and Dynamic Phase-encoding

2.6.1 Theory

Until now, dynamic encoding schemes changed the realization of the random variables from iteration to iteration to beat down the crosstalk term. In contrast, static encoding of shot gathers is the only option for field data where all of the shots are excited at nearly the same time and the blended wavefields are recorded as encoded supergathers. The excitation times or polarities of the shots are varied just once in the field trial, but cannot easily be changed after the data are recorded. In this case there should be no reduction of crosstalk by iterative stacking migration because the encoding function cannot be changed after each iteration.

However, equation 2.19 suggests that such supergathers can be least-squares migrated by an iterative steepest descent method, where the crosstalk strength is reduced by iterations. If the data are not encoded at each iteration then the value of the encoding function $c_{ij}^{(k)}$ in equation 2.19 is independent of the iteration k index, but still depends on the i and j indices. This suggests that the summations over i and j can act as proxies for ensemble averaging and reduce the crosstalk term under favorable conditions[17]. This reduction in crosstalk noise is expected to be not as great as that for dynamic encoding where the proxy for ensemble averaging is the summation over both the k and the i, j indices.

[17]In contrast, the crosstalk term in the iterative stacking equation 2.15 always grows as $O(k)$ for static encoding.

2.6.2 Numerical Tests

The hypothesis about the worse reduction in crosstalk noise by static LSM compared to dynamic LSM is validated by numerical tests on salt model data. Figure 2.10 shows that dynamic phase-encoding produces better migration images than the static counterpart for the same number of iterations. At 300 iterations of steepest descent, the migration image with dynamic phase-encoding comes very close to conventional LSM. In terms of residual model error, the final image with dynamic phase-encoding outperforms the static counterpart, as shown in Figure 2.11. If dynamic encoding is too expensive, a hybrid strategy is to use static encoding for a large number of iterations, and then reset with a new realization of the encoding functions. This should provide better crosstalk reduction than static LSM, but not as good as dynamic LSM.

2.7 Discussion and Conclusions

Formulas are presented that relate the strength of the crosstalk term in supergather migration to the covariance of the phase-encoding function. Analysis suggests that crosstalk noise in supergather migration is exponentially reduced with increasing values of static variance and frequencies. Source statics coupled to receiver statics, location statics, and amplitude statics can further reduce the strength of the crosstalk term. These formulas are also relevant in predicting crosstalk reduction for waveform inversion and least-squares migration of encoded supergathers because their crosstalk terms are identical to the one in this chapter.

The formula for the crosstalk damping coefficient suggests that the variance of the time static should be greater than the dominant period of the source wavelet. Therefore, the statics variance should increase with lower frequencies in a multiscale inversion strategy. Both theory and simulations show that on average encoding with both time and polarity statics is superior to encoding with just one of these encoding strategies.

Signal-to-noise (SNR) estimates show that $SNR^{stand.} = \sqrt{GS}$ for a standard migration image and $SNR^{super} = \sqrt{GI}$ for an image computed by migrating a phase-encoded supergather; here, G is the number of traces in a shot gather, I is the number of stacking iterations in the supergather, and S is the number of encoded/blended shot gathers that comprise the supergather. The importance of these formulas is that they provide a precise understanding of the relationship between different phase encoding strategies and image quality.

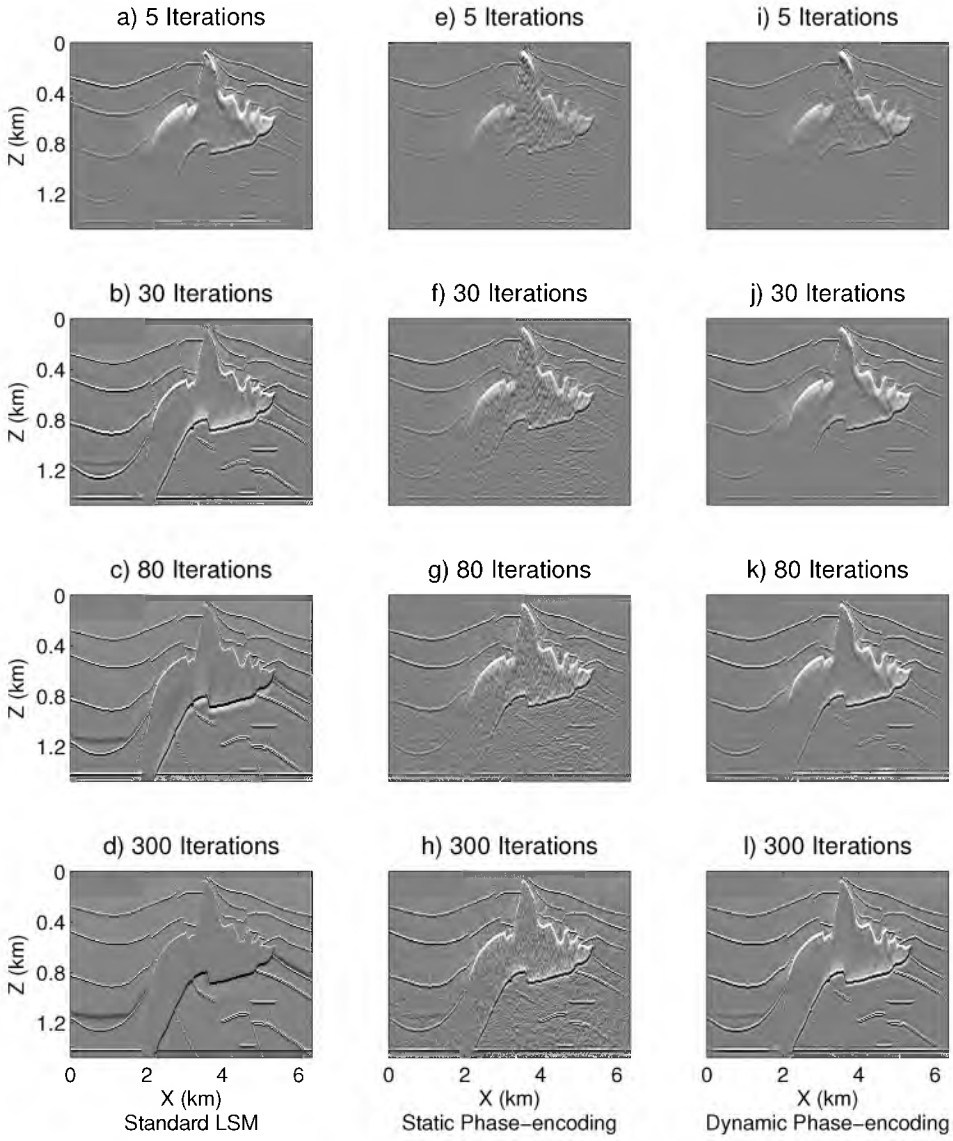

Figure 2.10. Comparison of standard, static phase-encoded and dynamic phase-encoded LSM images: a) through d) show the standard LSM images after 5, 30, 80 and 300 iterations; e) through h) illustrate the static phase-encoded LSM images after 5, 30, 80 and 300 iterations of a 320-shot supergather; while i) through l) present the dynamic phase encoding LSM images after 5, 30, 80 and 300 iterations of a 320-shot supergather. All input data are associated with the SEG/EAGE salt model and a source Ricker wavelet with a peak frequency of 50 Hz.

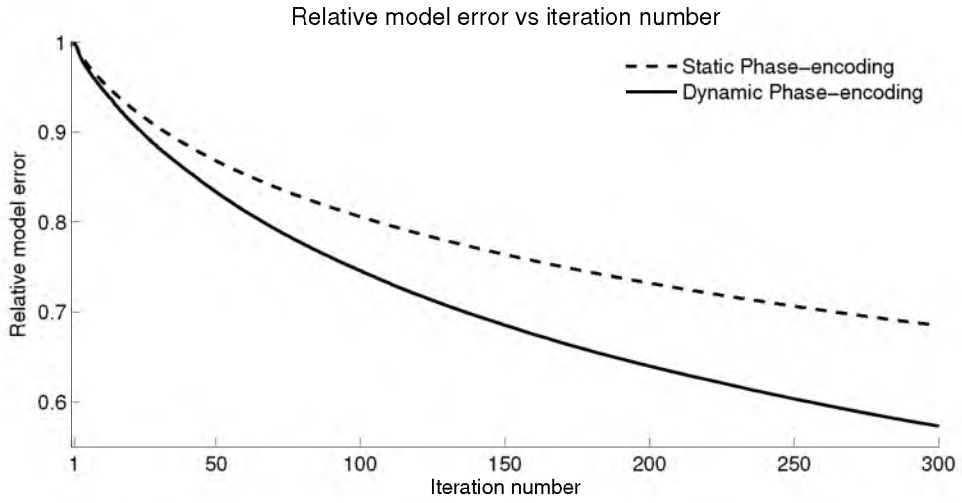

Figure 2.11. The solid line indicates the measured relative model error of least-squares migration with dynamic phase-encoding source time shift, while the dashed line indicates the one with static phase-encoding source time shift. Input data set consists of a 320-shot supergather associated with the SEG salt model, the input source wavelet is a Ricker wavelet with the peak frequency at 50 Hz.

CHAPTER 3

MULTISOURCE LEAST-SQUARES KIRCHHOFF MIGRATION

Least-squares migration (LSM) has been shown to produce high quality migration images, but its computational cost is too high for practical imaging. The multisource least-squares migration (MLSM) can increase the computational efficiency by utilizing the phase-encoding technique (see in section 2.3.2). In this chapter, the MLSM algorithm is implemented with the Kirchhoff migration operator. The phase-encoding functions can be kept the same or vary in each iteration or every few iterations, which correspond to static encoding, dynamic encoding and hybrid encoding, respectively. Both synthetic and field data results show that the multisource least-squres Kirchhoff migration images are of better quality than standard Kirchhoff migration images. The computational cost of Kirchhoff MLSM is about the same as standard LSM, but its IO cost is significantly decreased. It is also shown that dynamic encoding is more efficient in crosstalk noise reduction than static or hybrid encoding, and this validates the theory in section 2.6.

3.1 Introduction

It has been shown that least-squares migration (Nemeth et al., 1999; Duquet et al., 2000) can improve resolution of the migration images and suppress migration artifacts. However, one of the drawbacks of LSM is its high computational cost. Romero et al. (2000) proposed a blended source method by encoding and stacking different shot gathers into a supergather. The blended source data were formed by phase encoding each shot gather and stacking the shot gathers together to get a supergather. Dai and Schuster (2009) and Dai et al. (2011) adapted LSM to blended source data, which I now define as multisource least-squares migration procedure. This algorithm is applicable to Kirchhoff migration (Dai et al., 2011), wave-equation migration (Huang and Schuster, 2012) and reverse time migration (Dai et al., 2012). Schuster et al. (2011) provide rigorous formulas for predicting

the level of crosstalk noise as a function of the encoding parameters.

Application of wave equation MLSM to marine streamer data is hampered by the variable location of receivers for each shot. Huang and Schuster (2012) proposed a frequency division multiplexing strategy with multisource least-squares wave-equation migration. However, this procedure is not required for Kirchhoff based MLSM, which can be applied to marine streamer data without any restrictions on the acquisition geometry.

This chapter is organized into the following three sections. The first part reviews the theory of Kirchhoff migration (KM) and least-squares Kirchhoff migration, and presents the theory of multisource Kirchhoff migration and multisource least-squares Kirchhoff migration. The next section presents synthetic and field data results that demonstrate the efficiency and effectiveness of the MLSM algorithm. Finally, a summary is provided.

3.2 Theory

In this section, I review the theory of Kirchhoff (also known as diffraction stack) modeling and migration, and standard least-squares Kirchhoff migration. Then, I introduce the theory of multisource Kirchhoff migration and multisource least-squares Kirchhoff migration.

3.2.1 Kirchhoff Modeling and Migration

Seismic reflection data can be related to the earth model m by forward modeling (also known as the diffraction stack modeling (Claerbout, 1992)) as

$$d(\mathbf{r},t|\mathbf{s},0) = \int m(\mathbf{x})W(t) * G(\mathbf{r},t|\mathbf{x},0) * \ddot{G}(\mathbf{x},t|\mathbf{s},0)d\mathbf{x}, \tag{3.1}$$

where $G(\mathbf{x},t|\mathbf{s},0)$ and $G(\mathbf{r},t|\mathbf{x},0)$ are Green's functions for the source at $\mathbf{s}$ and the scattering point $\mathbf{x}$ and the receiver at $\mathbf{r}$, receptively; and $W(t)$ is the source wavelet. Under the high frequency approximation, the asymptotic Green's function is

$$G(\mathbf{x},t|\mathbf{x}',0) = \frac{\delta(t-\tau_{\mathbf{x}\mathbf{x}'})}{A_{\mathbf{x}\mathbf{x}'}}, \tag{3.2}$$

where $\tau_{\mathbf{x}\mathbf{x}'}$ is the travel time between two points $\mathbf{x}$ and $\mathbf{x}'$, and $A_{\mathbf{x}\mathbf{x}'}$ is the geometric spreading term accounts for the amplitude. Substituting equation 3.2 into equation 3.1 and assuming $W(t) = \delta(t)$ yields:

$$d(\mathbf{r},t|\mathbf{s},0) = \int m(\mathbf{x})\frac{\delta(t-\tau_{\mathbf{x}\mathbf{s}}-\tau_{\mathbf{r}\mathbf{x}})}{A_{\mathbf{x}\mathbf{s}}A_{\mathbf{r}\mathbf{x}}}d\mathbf{x}. \tag{3.3}$$

Its adjoint operation (known as diffraction stack migration or Kirchhoff migration) can be expressed as

$$\tilde{m}(\mathbf{x}') = \int d\mathbf{s}\int d\mathbf{r}\int d(\mathbf{r},t|\mathbf{s},0)\frac{\delta(t-\tau_{\mathbf{x}'\mathbf{s}}-\tau_{\mathbf{r}\mathbf{x}'})}{A_{\mathbf{x}'\mathbf{s}}A_{\mathbf{r}\mathbf{x}'}}dt, \tag{3.4}$$

here, $\mathbf{x}'$ is the trial image point. Equations 3.3 and 3.4 can be written as

$$\mathbf{d}_s = \mathbf{L}_s\mathbf{m}, \tag{3.5}$$

$$\hat{\mathbf{m}} = \mathbf{L}_s^{\dagger}\mathbf{d}_s. \tag{3.6}$$

and for multiple shots,

$$\mathbf{d} = \begin{bmatrix}\mathbf{d}_1\\ \mathbf{d}_2\\ \vdots\\ \mathbf{d}_S\end{bmatrix} = \begin{bmatrix}\mathbf{L}_1\\ \mathbf{L}_2\\ \vdots\\ \mathbf{L}_S\end{bmatrix}\mathbf{m} = \sum_{i=1}^{S}\mathbf{L}_i\mathbf{m} = \mathbf{L}\mathbf{m}, \tag{3.7}$$

$$\hat{\mathbf{m}} = \begin{bmatrix}\mathbf{L}_1^{\dagger} & \mathbf{L}_2^{\dagger} & \cdots & \mathbf{L}_S^{\dagger}\end{bmatrix}\begin{bmatrix}\mathbf{d}_1\\ \mathbf{d}_2\\ \vdots\\ \mathbf{d}_S\end{bmatrix} = \sum_{i=1}^{S}\mathbf{L}_i^{\dagger}\mathbf{d}_i = \mathbf{L}^{\dagger}\mathbf{d}. \tag{3.8}$$

3.2.2 Least-squares Kirchhoff Migration

Equation 3.8 can also be interpreted as the first iteration of linear inversion with a misfit function of

$$f(\hat{\mathbf{m}}) = \frac{1}{2}\|\mathbf{L}\hat{\mathbf{m}} - \mathbf{d}\|_2^2 + \mathbf{R}, \tag{3.9}$$

where the Hessian is approximated as a diagonal matrix. Here, $\mathbf{R}$ is a regularization term (Nemeth et al., 1999; Wang and Sacchi, 2007). This approximation is violated when the data are incomplete (Nemeth et al., 1999), which is usually true for practical imaging, so the migration image will be blurred by migration artifacts.

To overcome this blurring effect, least-squares Kirchhoff migration is introduced (Nemeth et al., 1999; Duquet et al., 2000) to find a solution that can minimize equation 3.9. Given the observed data $\mathbf{d}$, Kirchhoff modeling operator $\mathbf{L}$ and migration operator $\mathbf{L}^{\dagger}$, a preconditioned conjugate gradient implementation can be expressed as:

1. calculate the data residual at the k th iteration : $\delta\mathbf{d}^{(k)} = \mathbf{L}\hat{\mathbf{m}}^{(k)} - \mathbf{d}$;
2. get the gradient of $k+1$ th iteration: $\mathbf{g}^{(k+1)} = \mathbf{L}^{\dagger}\delta\mathbf{d}^{(k)} + \frac{\partial\mathbf{R}}{\partial\hat{\mathbf{m}}}$;
3. calculate the conjugate direction: $\mathbf{z}^{(k+1)} = \mathbf{P}\mathbf{g}^{(k+1)} + \beta\mathbf{z}^{(\mathbf{k})}$, where $\mathbf{P}$ is a preconditioner and $\beta = \frac{\langle\mathbf{g}^{(k+1)},\mathbf{P}\mathbf{g}^{(k+1)}\rangle}{\langle\mathbf{g}^{(k)},\mathbf{P}\mathbf{g}^{(k)}\rangle}$;
4. calculate the step length: $\alpha = \frac{\langle\mathbf{z}^{(k+1)},\mathbf{g}^{k+1}\rangle}{\langle\mathbf{L}\mathbf{z}^{(k+1)},\mathbf{L}\mathbf{z}^{(k+1)}\rangle + \ (term\ associated\ with\ \mathbf{R})}$;
5. update the model: $\hat{\mathbf{m}}^{(k+1)} = \hat{\mathbf{m}}^{(k)} - \alpha\mathbf{z}^{(k+1)}$;

6. repeat steps 1 - 5 until an acceptable data residual is achieved or the iteration reaches the user-defined maximum iteration number.

3.2.3 Multisource Kirchhoff Modeling and Migration

Multisource modeling and migration can be expressed by equations 2.11 and 2.13. As shown in Figure 3.1, two shot gathers (b and c) with different recording aperture are encoded by the receiver-side encoding function[1] (contain both polarity and time-shift statics) to form a supergather (h). Since the recording aperture is known, these two shot gathers can be decoded[2] correctly (Figure 3.1i and j).

Figure 3.2 depicts the key steps for applying multisource Kirchhoff migration to marine streamer data. Two distinct shot gathers with different recording apertures (Figure 3.2a) are first phase-encoded (Figure 3.2b) then stacked to generate a supergather (Figure 3.2c). In Kirchhoff migration, each trace is decoded for the correct shot and receiver locations before migration. For example, if a trace is encoded by time shifting 1 s, then that trace is decoded by a -1 s time shift (Figure 3.2d). In Figure 3.2e, the event in the blue solid circle is smeared to the model space (solid blue ellipse) with the travel time from associated source and receiver of CSG1, while the event in the red solid circle is smeared with the travel time from the associated source and receiver of CSG2. Events with dashed lines are smeared with dashed ellipses. Therefore, the solid lines indicate the migration artifacts and the dashed lines indicate the crosstalk noise.

3.2.4 Multisource Least-squares Kirchhoff Migration

The multisource least-squares Kirchhoff migration algorithm was introduced in section 2.3.2. The procedure for MLSM is similar to LSM, except:

1. The Kirchhoff modeling and migration operator $\mathbf{L}$ and $\mathbf{L}^\dagger$ are replaced with the multisource Kirchhoff modeling and migration operators $\mathcal{L}$ and $\mathcal{L}^\dagger$, introduced in

---

[1]The ith shot gather $\mathbf{d}_i$ contains G receivers, and can be expressed $\mathbf{d}_i = \sum_{j=1}^{G} d_{i,j}$. Each trace is encoded with its own phase-encoding function $N_{i,j}$, so the ith encoded shot gather is $\tilde{\mathbf{d}}_i = \sum_{j=1}^{G} N_{i,j} d_{i,j}$. By stacking S shots, the supergather can be expressed as $\tilde{\mathbf{d}} = \sum_{i=1}^{S}\sum_{j=1}^{G} N_{i,j} d_{i,j}$.

[2]Decoding is the procedure of applying the adjoint operation $\sum_{j=1}^{G} N_{i,j}^*$ to the supergather $\tilde{\mathbf{d}}$ to give the ith decoded shot gather $\mathbf{d}_i' = \sum_{j=1}^{G} d_{i,j} + \sum_{i'\neq i}^{S}\sum_{i'=1}^{S}\sum_{j=1}^{G} \langle N_{i',j}^* N_{i,j}\rangle d_{i',j}$. Applying the migration operator associated with ith shot position to the first term will give the standard Kirchhoff migration image $\hat{\mathbf{m}}_i$ of ith shot, and the second term will result in the crosstalk noise term $\tilde{\mathbf{c}}_i$.

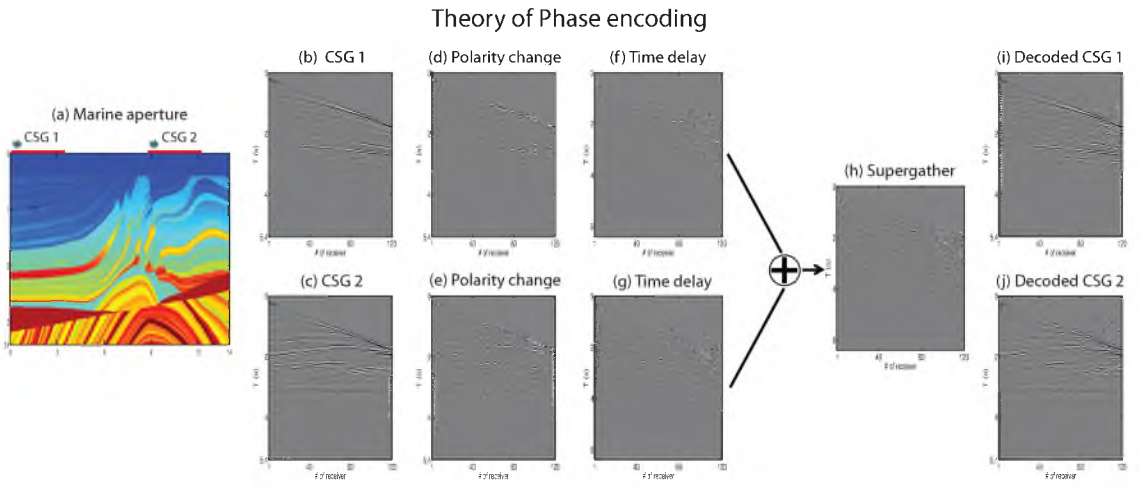

Figure 3.1. Steps for creating a phase-encoded supergather and decoding. (a) shows the receiver and shot positions associated with two distinct shot gathers; (b), (d) and (f) show the original, polarity encoded and receiver-side time shifted encoded common shot gather (CSG) 1, while (c), (e) and (g) are for CSG 2; (h) is the supergather after summing of (f) and (g); and (i) and (j) are the decoded CSGs for migration.

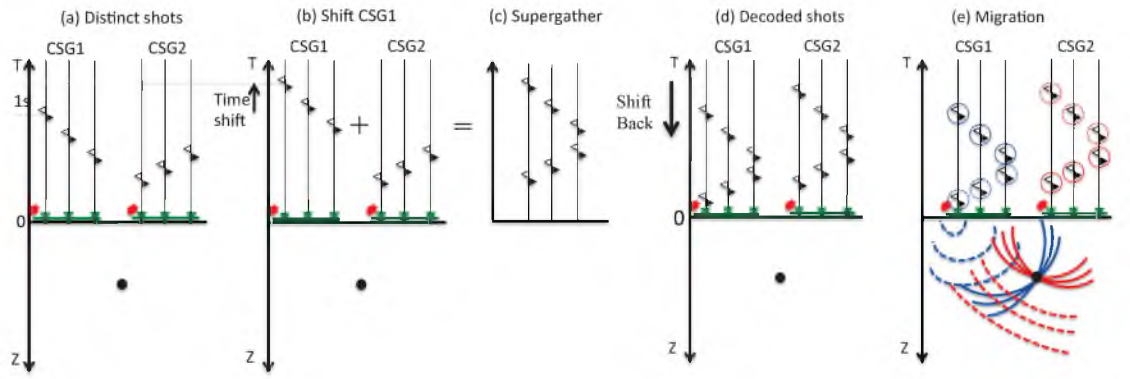

Figure 3.2. Illustration of steps for multisource Kirchhoff migration. (a) shows the single shot CSGs with a marine streamer acquisition aperture, receiver positions are different for each shot, (b) shows phase-encoding which shifts CSG1 1 second later, (c) shows the supergather data generated by stacking traces from CSG1 and CSG2, (d) shows the decoded shot, which shifts the supergather trace 1 second back and (e) shows the migration of the decoded CSGs with the associated source and receiver positions.

section 3.2.3.

2. Seismic data $\mathbf{d}$ are replaced with the supergather data $\mathcal{D}$

$$\mathcal{D} = \sum_{i_{sub-super}=1}^{N_{sub-super}} \mathbf{N}_i^{(k)} \mathbf{d}_i, \tag{3.10}$$

where, $\mathbf{N}_i^{(k)}$ is the phase-encoding function for shot $\mathbf{d}_i$ at the kth iteration.

3. The phase-encoding function $\mathbf{N}_i^{(k)}$ can be chosen to be a) kept the same for all iterations, b) vary at each iteration, and c) vary for every few iterations, which correspond to static, dynamic and hybrid encoding, respectively.

In this study, we use the conjugate gradient (CG) method for static encoding, steepest descent (SD) method for dynamic encoding because the conjugacy is ruined by changing the encoding function during each iteration, and SD+CG method (reset the conjugate direction at each re-encoding iteration) for hybrid encoding.

3.3 Numerical Results

MLSM with the Kirchhoff operator is tested on both synthetic and field data. The goal is to compare the image quality, the CPU expense, the IO demand and the memory costs for KM, LSM and MLSM.

3.3.1 Marmousi2 Model with FD Data

The synthetic data are generated by a finite-difference solution to the acoustic wave equation for the 2D Marmousi2 model (Figure 3.3) with the marine streamer acquisition geometry. The model size is 1401 $\times$ 341, and the grid interval is 10 m, with 192 shot gathers having a shot interval of 60 m, and each shot gather is recorded with a 2.4 km long streamer of 120 receivers with the receiver interval of 40 m. A Ricker wavelet with a peak frequency of 20 Hz is used as the source wavelet. Figures 3.4 and 3.5 show the standard Kirchhoff and least-squares migration images after 15 iterations, respectively. Three zoom views are shown in Figures 3.6-3.8. Compared with standard Kirchhoff migration, LSM decreases the migration artifacts (especially in the shallow depth), balances the amplitudes, and increases the resolution.

For the MLSM, 16 supergathers are generated with both the source polarity and source time statics encoding (with the standard deviation $\sigma = 0.3s$ of the source static shifts), and each supergather consists of 12 stacked shot gathers. Figure 3.9 shows the MLSM result with static encoding after 15 iterations. Figure 3.10 shows the MLSM result with hybrid encoding

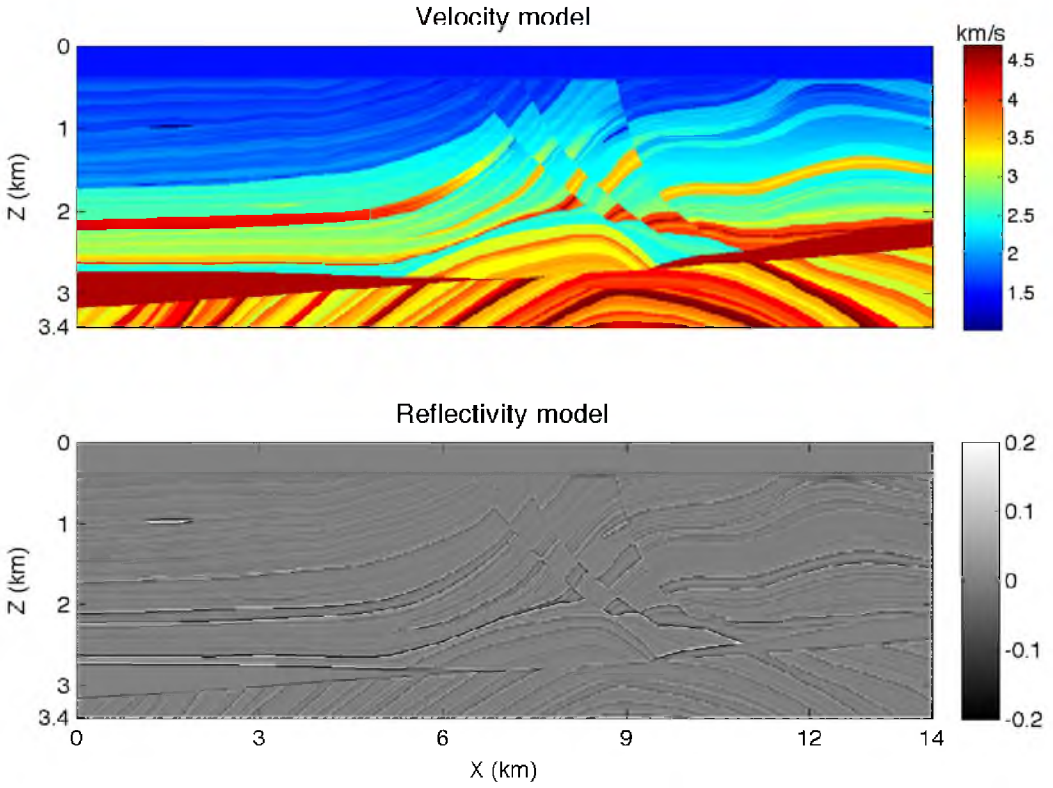

Figure 3.3. 2D Marmousi2 model: a) velocity model and b) reflectivity model. Synthetic data are generated by a FD solution to the acoustic wave equation with a Ricker wavelet at the peak frequency of 20 Hz.

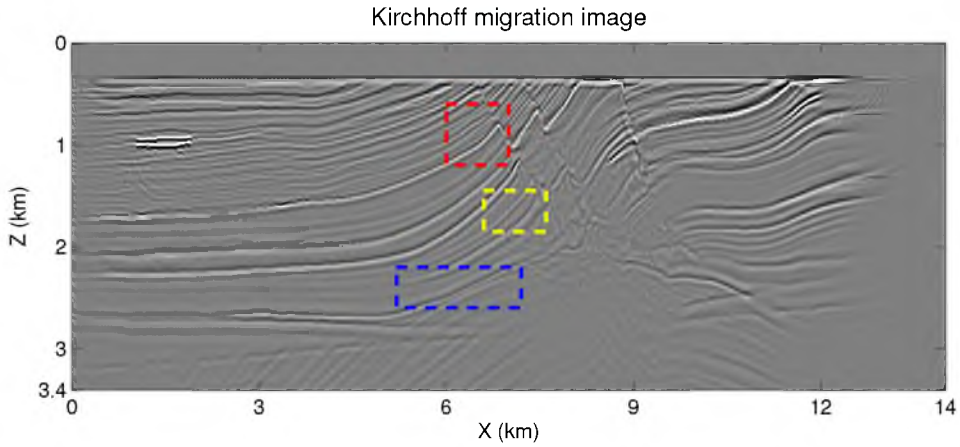

Figure 3.4. Standard Kirchhoff migration image of the 2D Marmousi2 model.

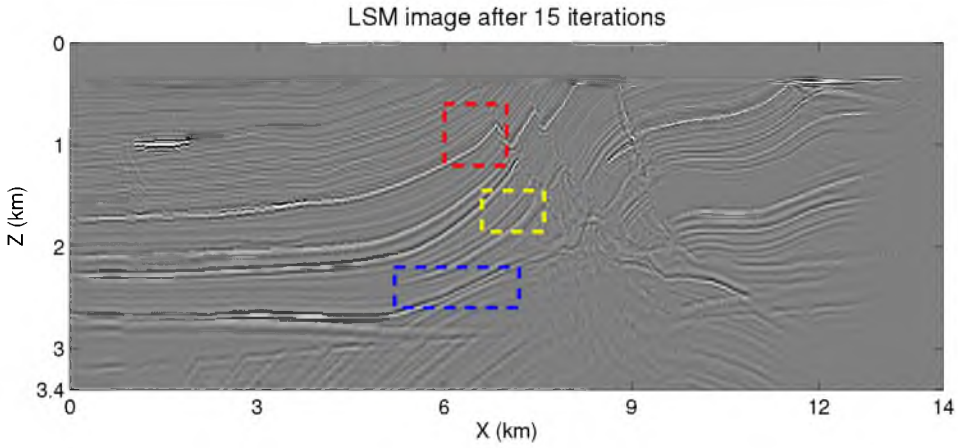

Figure 3.5. Least-squares migration image after 15 iterations of the 2D Marmousi2 model.

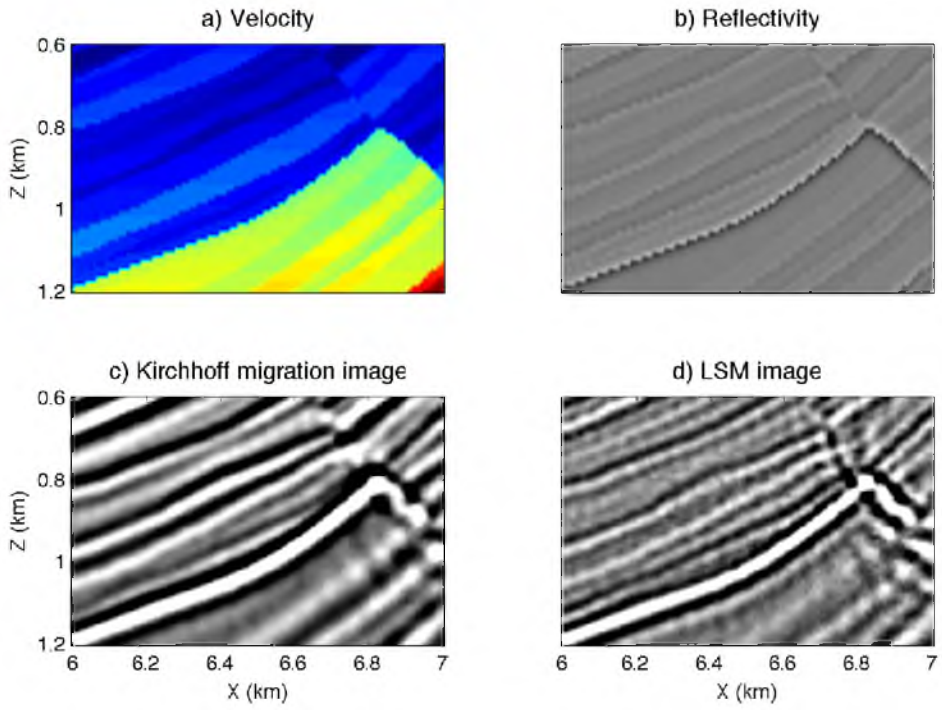

Figure 3.6. Zoom views of red boxes in Figures 3.3-3.5: a) velocity model, b) reflectivity model, c) Kirchhoff migration and d) least-squares migration.

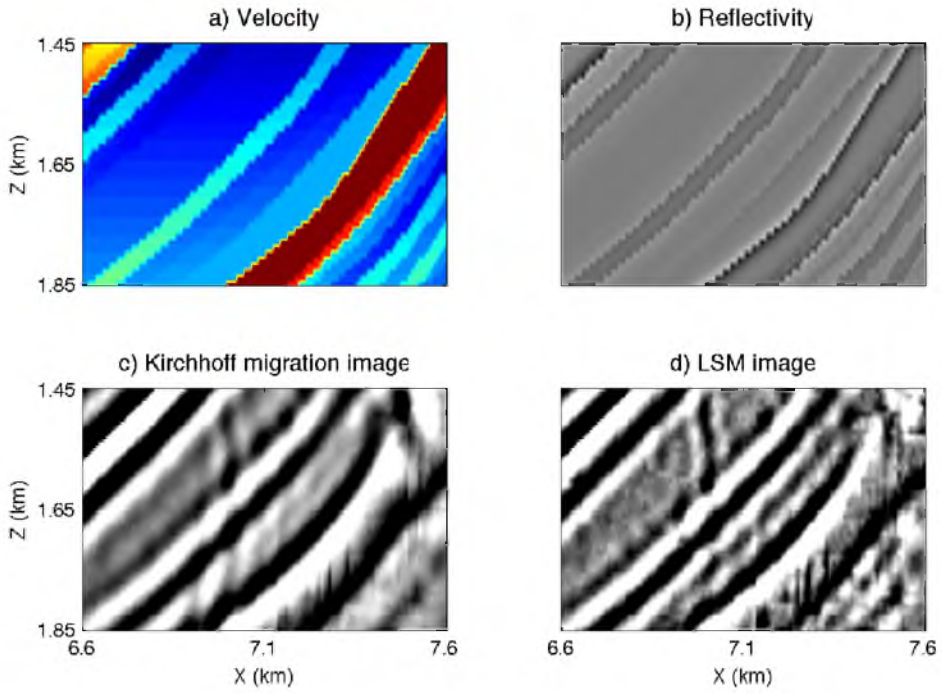

Figure 3.7. Zoom views of yellow boxes in Figures 3.3-3.5: a) velocity model, b) reflectivity model, c) Kirchhoff migration and d) least-squares migration.

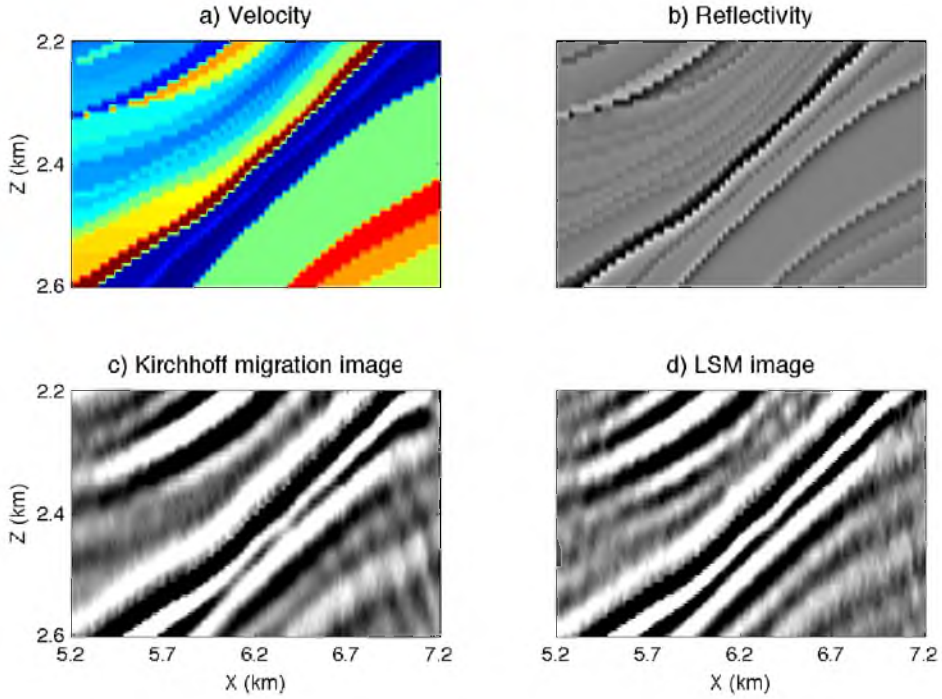

Figure 3.8. Zoom views of blue boxes in Figures 3.3-3.5: a) velocity model, b) reflectivity model, c) Kirchhoff migration and d) least-squares migration.

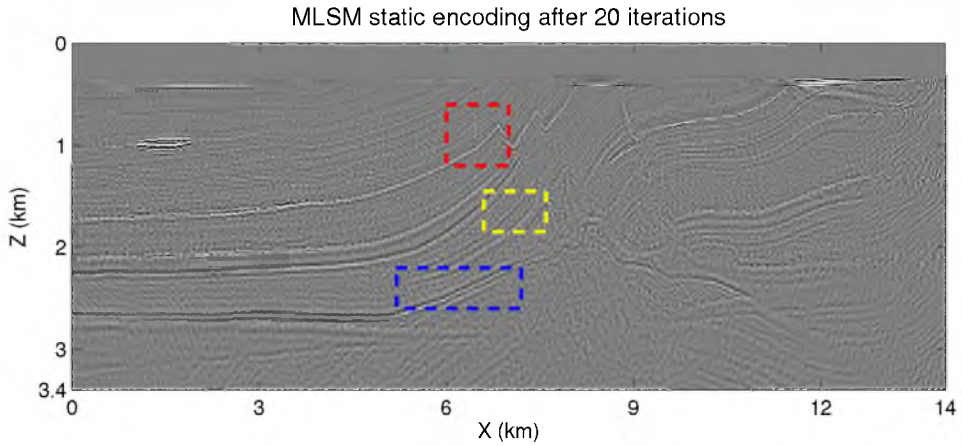

Figure 3.9. Multisource least-squares migration image with static phase encoding after 20 iterations.

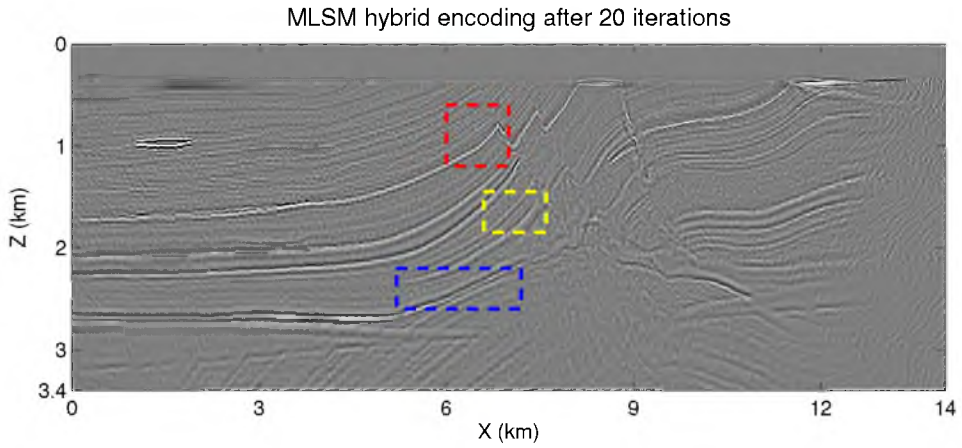

Figure 3.10. Multisource least-squares migration image with hybrid phase encoding after 20 iterations.

(change the encoding function for every 5 iterations) after 20 iterations. Figure 3.11 shows the MLSM result with dynamic encoding after 20 iterations. Figures 3.12-3.14 show the same zoom views for these three different encoding functions. Among the three images with different encoding function, the dynamic encoding results show the best quality image, which is close to that of single source LSM. The second best is the one generated by MLSM with hybrid encoding. If the IO cost is considered, then static encoding is the most efficient, and the second best is the hybrid encoding. Table 3.1 compares the costs of these methods. LSM and MLSM are compared at 15 iterations, the CPU, IO and memory cost for KM are defined as 1.0. For MLSM, the supergather data are are assumed to be small enough to be stored in physical memory.

3.3.2 Gulf of Mexico Marine Streamer Data

A Gulf of Mexico data set is tested with MLSM. The model size is 3401 by 401 gridpoints, with a gridpoint interval of 6.25 m, and Figure 3.15 shows the P-wave velocity model estimated from full waveform inversion (Boonyasiriwat et al., 2010). There are 496 shots with a shot interval of 37.5 m, and each shot shoots into a 6-km-long streamer, which contains 480 receivers with a receiver interval of 12.5 m. For each shot the nearest offset is 200 m. The preprocessing procedures contains: 1) 3D to 2D amplitude correction, 2) muting of the direct wave in common offset gather domain and 3) a bandpass filter with the filtering window of 10-15-70-75 Hz. The source wavelet is estimated by stacking the near-offset ocean-bottom reflections. Figure 3.16 shows the standard migration image and Figure 3.17 shows the least-squares migration image after 30 iterations. Two detailed areas (blue and yellow boxes in Figures 3.16 and 3.17) are magnified in Figure 3.18a-b and Figure 3.19a-b. The most significant improvements are that the resolution increases (as shown with the blue arrows) and the amplitude is more balanced (as shown by the green circles). The other advantage of LSM is the suppresion of migration artifacts due to the incomplete data, which is not extant in this example.

Thirty-two supergathers are generated from the total of 496 shot gathers, and each supergather consists of 15 or 16 shots. Figures 3.20 and 3.21 show the multisource least-squares migration image with static encoding and dynamic encoding, respectively, and both are stopped after 30 iterations. Comparing the standard LSM image (Figure 3.17), MLSM with dynamic encoding can provide almost the same quality image, while the quality of MLSM with static encoding is not acceptable due the large amount of crosstalk noise. The same two detailed areas are shown in Figures 3.18c-d and 3.19c-d, and it is shown that MLSM with dynamic encoding has the same advantages as LSM, and also provides

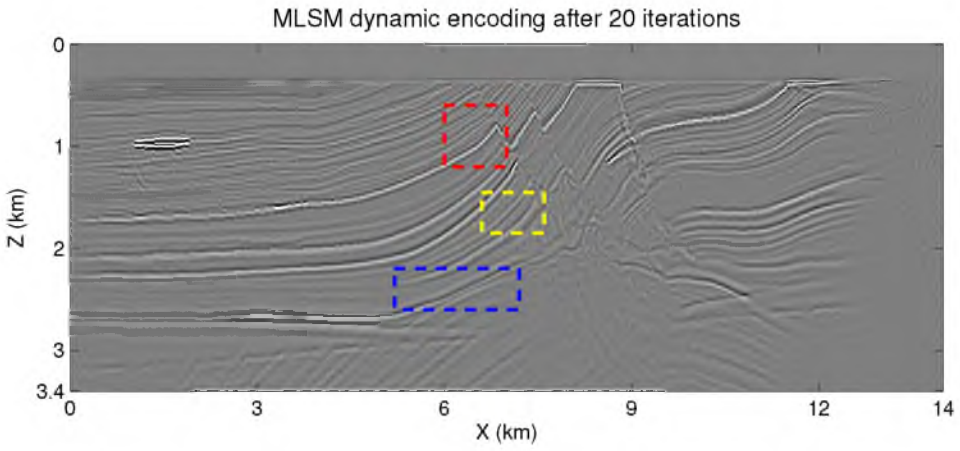

Figure 3.11. Multisource least-squares migration image with dynamic phase encoding after 20 iterations.

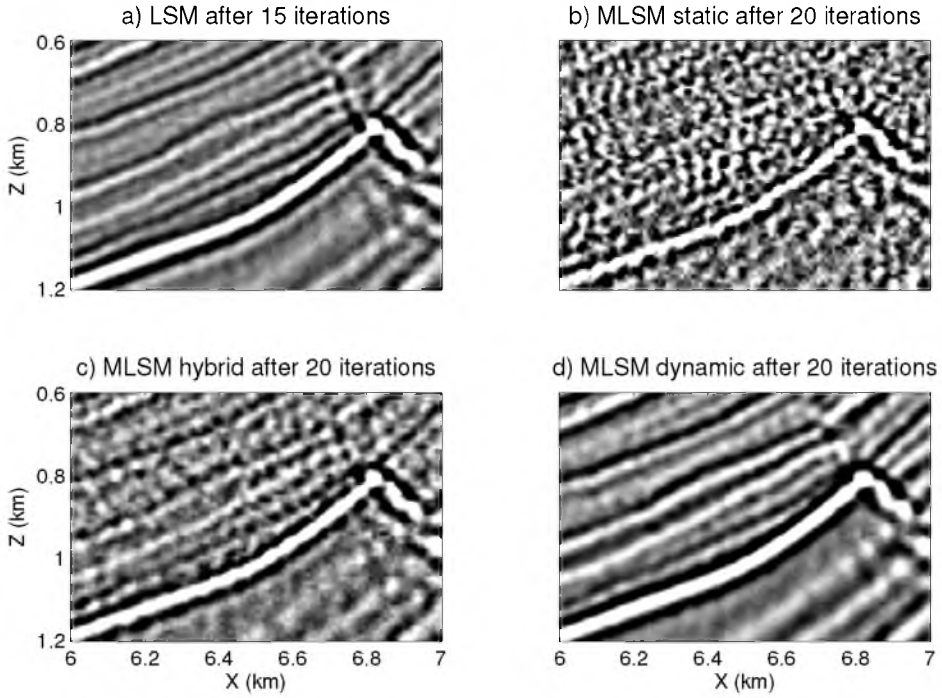

Figure 3.12. Zoom views of red boxes in Figures 3.5, 3.9, 3.10 and 3.11: a) standard LSM, b) MLSM with static encoding, c) MLSM with hybrid encoding and d) MLSM with dynamic encoding.

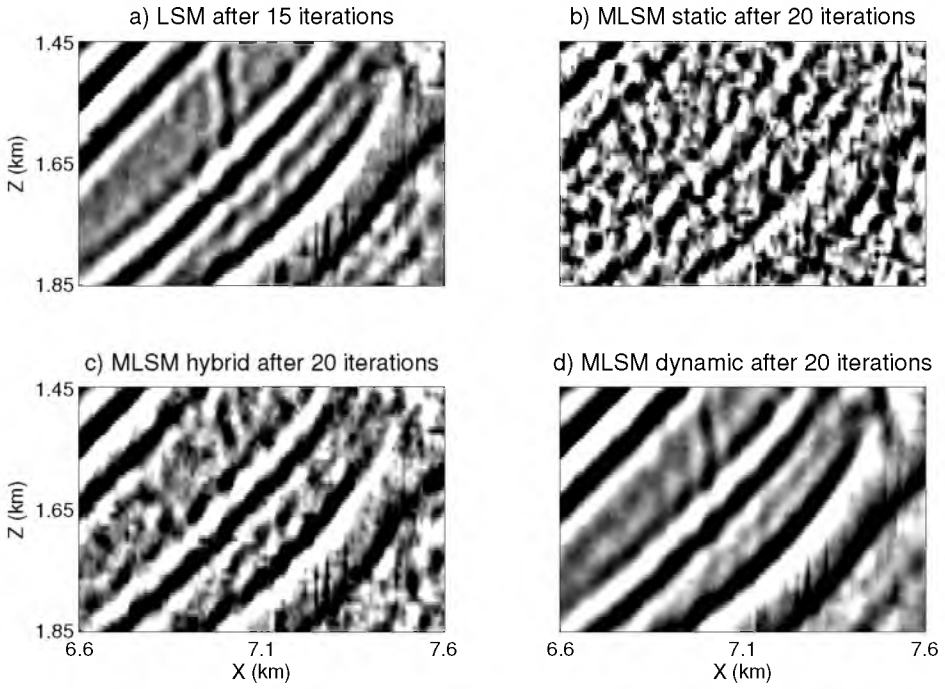

Figure 3.13. Zoom views of yellow boxes in Figures 3.5, 3.9, 3.10 and 3.11: a) standard LSM, b) MLSM with static encoding, c) MLSM with hybrid encoding and d) MLSM with dynamic encoding.

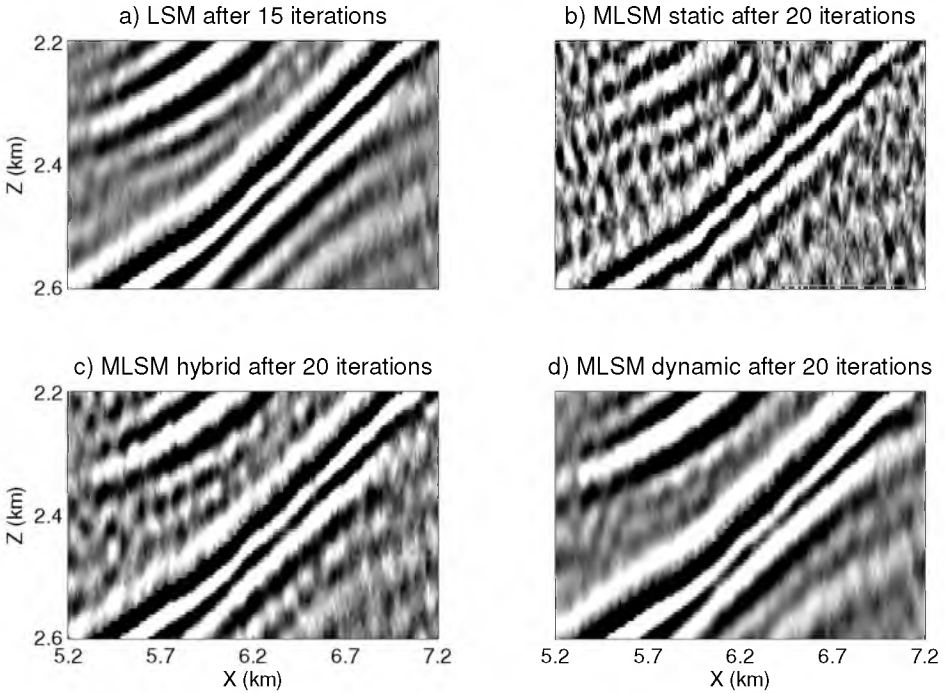

Figure 3.14. Zoom views of blue boxes in Figures 3.5, 3.9, 3.10 and 3.11: a) standard LSM, b) MLSM with static encoding, c) MLSM with hybrid encoding and d) MLSM with dynamic encoding.

Table 3.1. Comparison of image quality and CPU, IO and memory costs for KM, LSM and MLSM for the Marmousi2 synthetic data test. (LSM I does not store data in memory, but LSM II does.)

	Image quality	**CPU cost**	**IO cost**	**Memory cost**
KM	⋆	1.0	1.0	1.0
LSM I	⋆⋆⋆⋆⋆	31.0	15.0	1.0
LSM II	⋆⋆⋆⋆⋆	31.0	1.0	192.0
Static MLSM	⋆⋆	31.0	0.08	16.0
Hybrid MLSM	⋆⋆⋆	31.0	0.25	16.0
Dynamic MLSM	⋆⋆⋆⋆	31.0	1.25	16.0

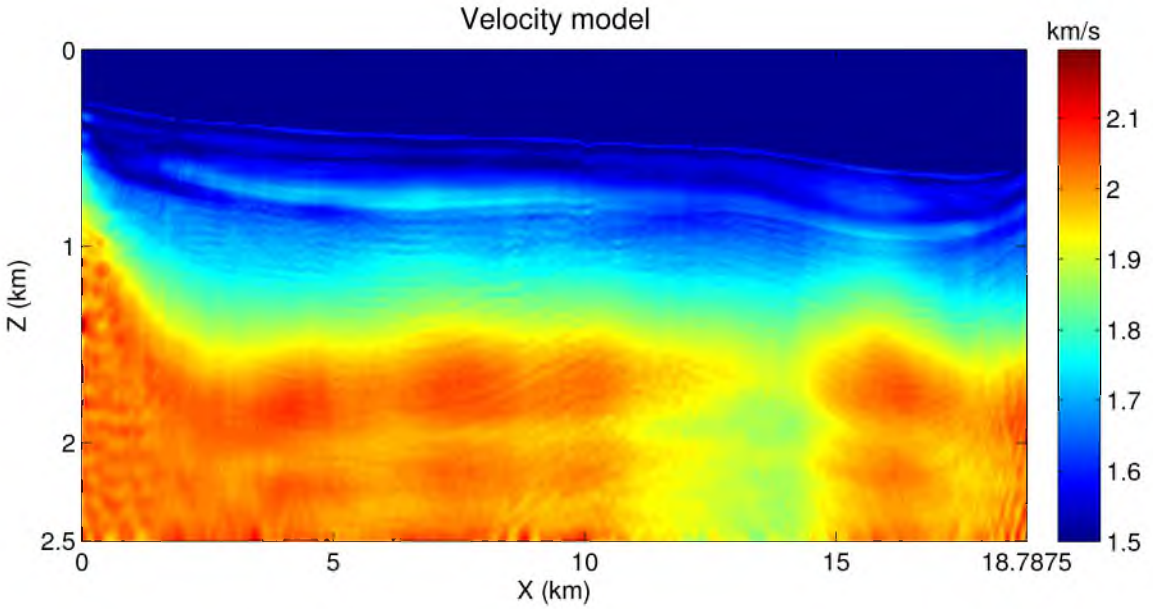

Figure 3.15. P-wave velocity model estimated of the Gulf of Mexico marine data from full waveform inversion.

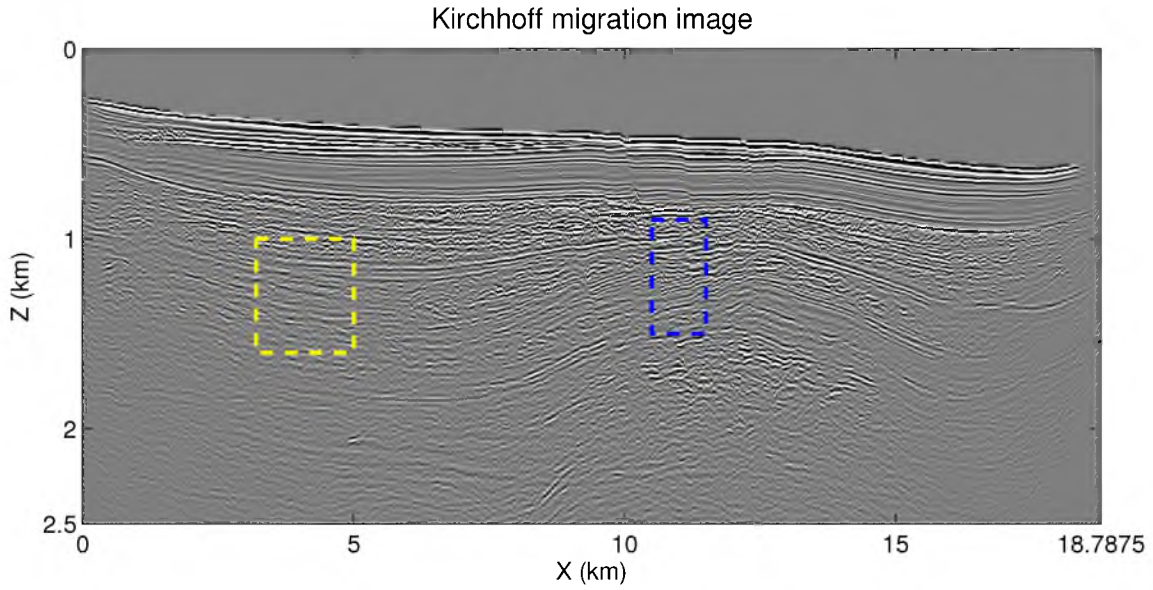

Figure 3.16. Standard Kirchhoff migration image of GOM marine data.

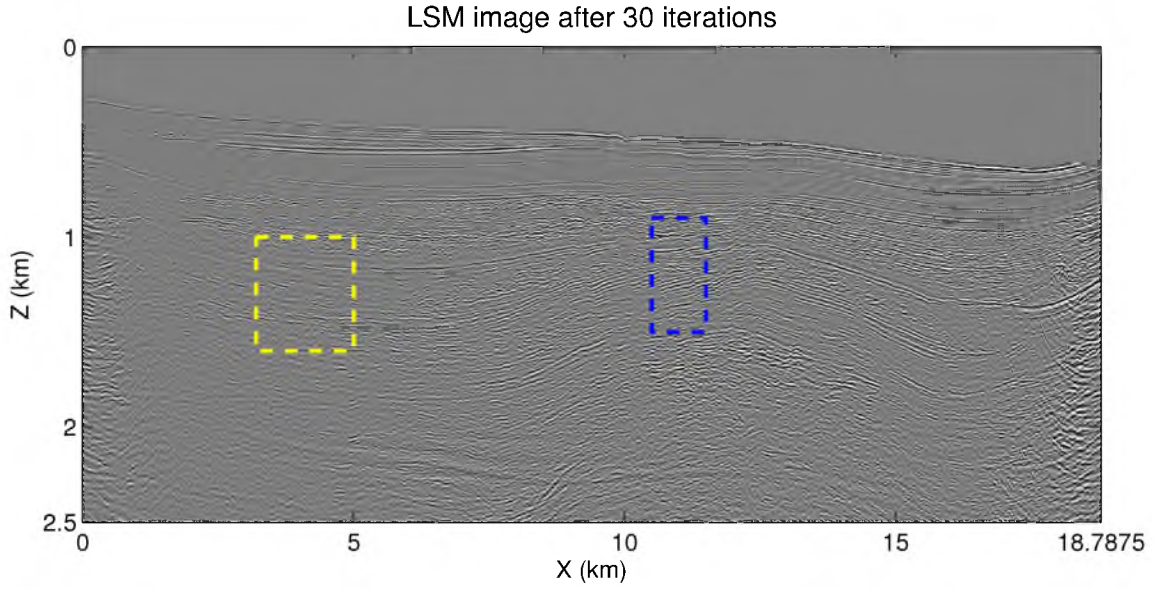

Figure 3.17. Standard least-squares Kirchhoff migration image of GOM marine data after 30 iterations.

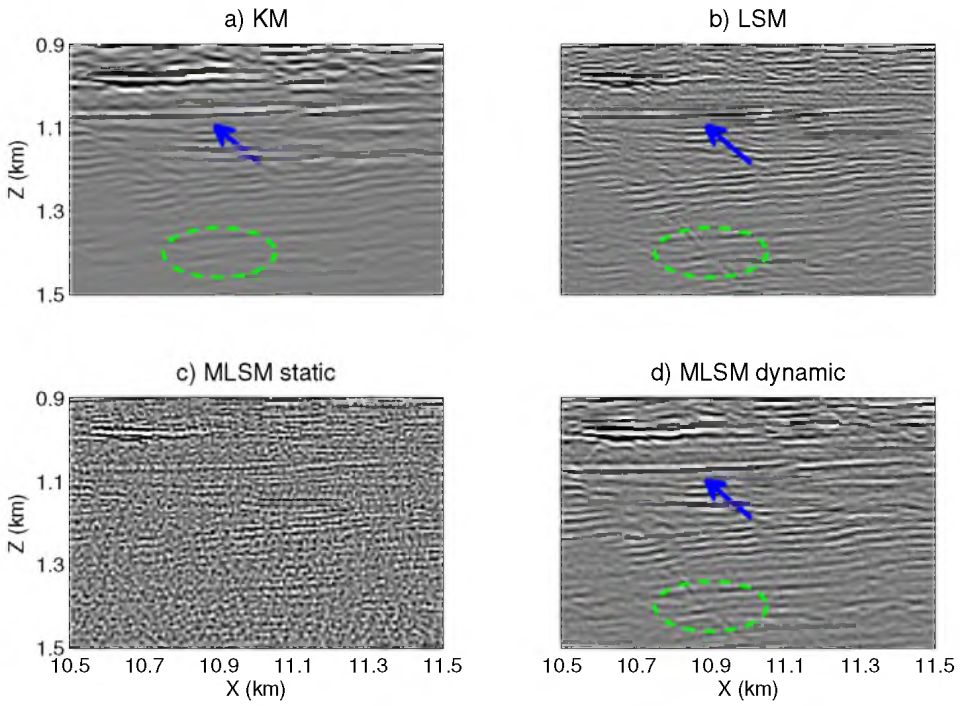

Figure 3.18. Zoom views of blue boxes in Figures 3.16-3.21: a) standard KM, b) standard LSM, c) MLSM with static encoding and d) MLSM with dynamic encoding.

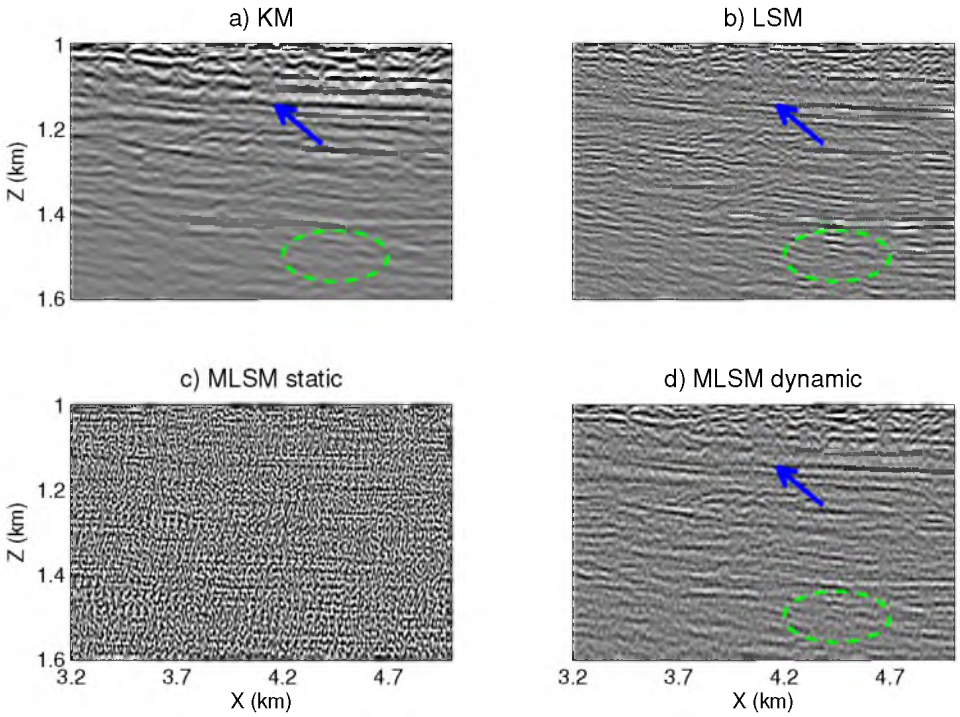

Figure 3.19. Zoom views of yellow boxes in Figures 3.16-3.21: a) standard KM, b) standard LSM, c) MLSM with static encoding and d) MLSM with dynamic encoding.

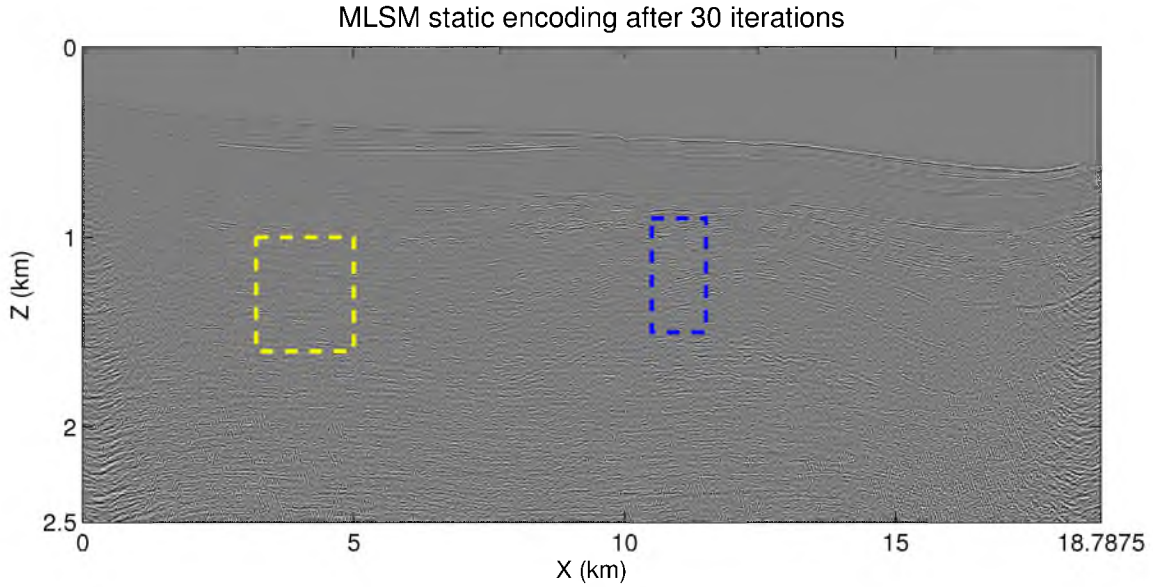

Figure 3.20. Multisource least-squares migration image with static phase-encoding of GOM marine data after 30 iterations.

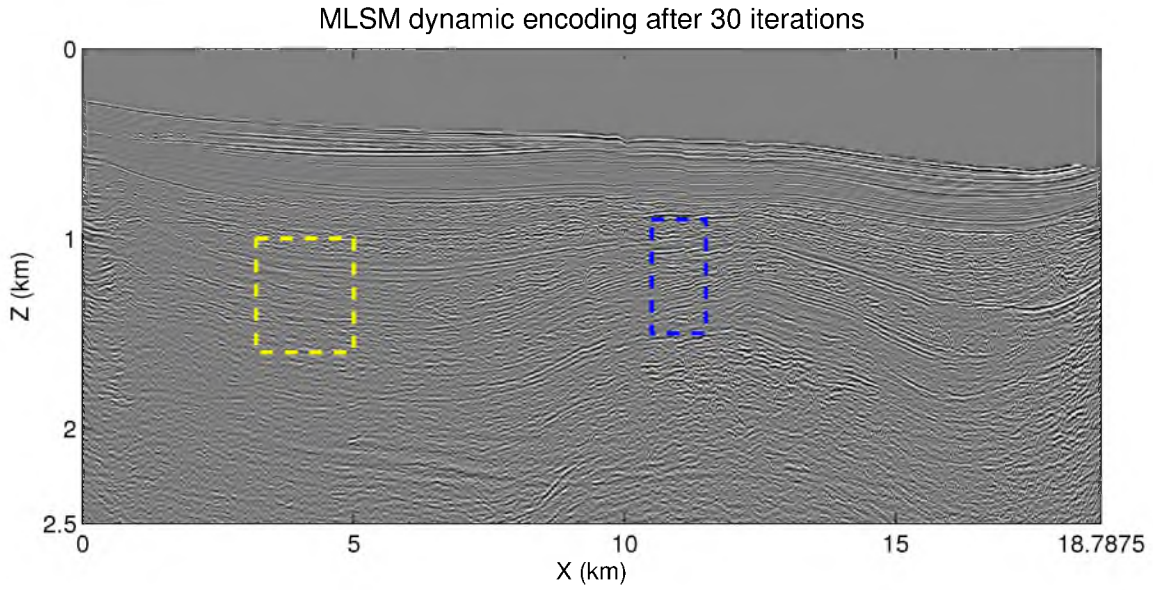

Figure 3.21. Multisource least-squares migration image with dynamic phase-encoding of GOM marine data after 30 iterations.

high quality images. Regarding the IO and memory costs, by stacking 496 shots into 32 supergathers, an IO saving of 92% is achieved for MLSM with dynamic encoding compared to standard LSM. However, the CPU cost is the same for LSM and MLSM.

3.4 Discussion and Conclusion

In this chapter, the multisource least-squares migration proposed in Chapter 2 is implemented with the Kirchhoff migration operator. Kirchhoff MLSM does not require the encoded shots to share the same receiver number and locations, thus it can be applied to marine streamer data. Numerical tests on the 2D Marmousi2 model, where data are generated by a finite difference solution to the acoustic wave equation, show that MLSM is robust and has high resolution. A Gulf of Mexico field data set is also tested with Kirchhoff MLSM. Both synthetic and field data tests show that MLSM can provide significantly higher quality images than that obtained by standard Kirchhoff migration. Compared to standard LSM, MLSM can significantly decrease the IO and memory cost, but not the CPU cost. To achieve a computational efficiency, a plane-wave encoding technique, which is a special case of multisource phase-encoding, will be introduced in Chapter 4 (for 2D case) and Chapter 5 (for 3D case). Field results also suggest that dynamic encoding achieves the best image quality, while static encoding fails to provide an acceptable image.

CHAPTER 4

2D PLANE-WAVE LEAST-SQUARES KIRCHHOFF MIGRATION

In this chapter, I present a new migration method that promises to give better quality images than standard Kirchhoff migration at significantly less cost than Kirchhoff least squares migration. I denote this method as plane-wave least-squares Kirchhoff migration (PWLSM). To account for the mispositioning of reflectors due to velocity model errors, a regularized plane-wave least-squares Kirchhoff migration (RPWLSM) is developed so that each plane-wave gather is migrated to give an individual migration image. A regularization term is included to encourage the similarities between the migration images with similar incidence angles. A 2D finite-difference plane-wave eikonal solver is developed to calculate the plane-wave travel times. Both synthetic and field results show the benefits of PWLSM and RPWLSM compared to Kirchhoff LSM: 1) PWLSM and RPWLSM can significantly reduce the CPU, IO and memory costs; 2) RPWLSM is more robust when the velocity model is inaccurate; 3) RPWLSM can provide high quality common image gathers (CIG), which can be used to correct the velocity model by migration velocity analysis. Aligning the migration reflectors before stacking by trim statics leads to more focused images.

4.1 Introduction

Least-squares migration (LSM) (Nemeth et al., 1999; Duquet et al., 2000) can produce high quality images but with high computational cost. As a partial remedy, Romero et al. (2000) proposed a blended source method for conventional migration by encoding and stacking different shot gathers into a supergather. This algorithm can be iteratively applied to LSM with Kirchhoff migration (Dai et al., 2011; Wang and Schuster, 2012), wave-equation migration (Huang and Schuster, 2012) and reverse time migration (Dai et al., 2012) to form the multisource least-squares migration (MLSM). The main problem with noniterative supergather migration is that the quality of the migration image will be degraded due to crosstalk noise. Here, the crosstalk noise is generated by the uncorrelated source and receiver

wavefields in the imaging condition. Dai et al. (2010, 2011), Huang and Schuster (2012), and Wang and Schuster (2012) showed that iterative least-squares migration with dynamic encoding incorporated into the supergather can suppress not only the migration artifacts, but also the crosstalk noise. The formulas for crosstalk noise reduction of MLSM are derived in Schuster et al. (2011).

Wave-equation MLSM can achieve high computational efficiency (Dai et al., 2011; Huang and Schuster, 2012) by modeling and migrating the supergather with one finite-difference solution to the wave equation for multiple sources. For Kirchhoff LSM, however, the computational cost is determined by the total number of traces, which cannot be reduced by the blended encoding of shot gathers. A linear time-shift phase encoding, which is identical to the tau-p transform (Zhang et al., 2005; Liu et al., 2006), can transform the shot-domain data to the plane-wave domain. By replacing the number of shots with a smaller number of ray parameters, a significant computational savings is achieved for plane-wave Kirchhoff migration (PWKM) (Akbar et al., 1996; Al-Saleh, 2002; Stoffa et al., 2006). In this chapter, I apply the plane-wave encoding technique to Kirchhoff LSM, and propose the algorithm of plane-wave least-squares Kirchhoff migration (PWLSM).

For Kirchhoff migration, one of the key steps is the calculation of travel times. In this chapter, I propose a new finite-difference solver to the 2D eikonal equation with the expanding plane-wavefront.

Another drawback of LSM is that it is more sensitive to the accuracy of the migration velocity model than conventional migration. When the velocity model contains large bulk errors, the migration images from different shots are inconsistent with each other, and simple stacking will blur the image and slow the convergence. To remedy this problem and increase the robustness of PWLSM, individual migration images are produced for different plane-wave gathers (Dai and Schuster, 2013; Wang et al., 2013). Since the images from two plane-wave gathers with slightly different incidence angles are similar, a regularization term is applied to encourage their similarities. This algorithm is denoted as regularized plane-wave least-squares migration (RPWLSM). Also, this method provides the opportunity for migration velocity analysis and trim-statics (Huang et al., 2014) with stacking to produce more focused images.

This chapter is organized into the following three sections. The first section presents the theory of PWLSM and RPWLSM. The next section presents synthetic and field data results that demonstrate the efficiency and effectiveness of PWLSM and RPWLSM. A summary is provided in the last section.

4.2 Theory

The theory of plane-wave encoding, plane-wave Kirchhoff modeling and migration is now presented. Then, the algorithms of PWLSM and RPWLSM are derived and a 2D finite-difference solver to the eikonal equation for plane-wave travel times is introduced.

4.2.1 2D Plane-wave Encoding

Plane-wave encoding can be defined as applying a linear time shift to the shot gathers and summing them together to form the response to an incident plane wave with a specified ray parameter p (Akbar et al., 1996; Zhang et al., 2005; Stoffa et al., 2006). For a 2D survey geometry, the encoding process is expressed as:

$$d(x_r, t; p) = \sum_{x_s} d(x_r, t; x_s) * \delta(t - p \cdot (x_s - x_0)), \tag{4.1}$$

where $d(x_r, t; x_s)$ is the shot-domain data associated with the source position x_s and receiver position x_r, x_0 is the reference source position[1](Zhang et al., 2005; Vigh and Starr, 2008), p is the ray parameter and $d(x_r, t; p)$ is the encoded plane-wave gather. As illustrated in Figure 4.1, the time shift $p \cdot (x_s - x_0)$ is a linear function of $x_s - x_0$, and p is defined as

$$p = \frac{sin\theta}{v}, \tag{4.2}$$

where θ is the surface shooting angle and v is the velocity at the source position.

4.2.2 Plane-wave Kirchhoff Modeling and Migration

Similar to conventional Kirchhoff modeling (seen in section 3.2.1), plane-wave Kirchhoff modeling associated with a specified p can be expressed as:

$$d(\mathbf{r}, t|p) = \int m(\mathbf{x}) \frac{\delta(t - \tau_{\mathbf{x},p} - \tau_{\mathbf{rx}})}{A_{\mathbf{rx}}} d\mathbf{x}, \tag{4.3}$$

where $m(\mathbf{x})$ is the earth model, $\tau_{\mathbf{rx}}$ is the travel time from $\mathbf{x}$ to the receiver at $\mathbf{r}$, $\tau_{\mathbf{x},p}$ is the travel time at $\mathbf{x}$ associated with the ray parameter p, and $A_{\mathbf{rx}}$ is the geometric spreading term from $\mathbf{x}$ to $\mathbf{r}$[2].

Its adjoint operation (denoted as plane-wave Kirchhoff migration) can be expressed as

$$\widehat{m}(\mathbf{x}') = \int d\mathbf{r} \int d(\mathbf{r}, t|p) \frac{\delta(t - \tau_{\mathbf{x}',p} - \tau_{\mathbf{rx}'})}{A_{\mathbf{rx}'}} dt, \tag{4.4}$$

where $\widehat{m}(\mathbf{x}')$ is the plane-wave Kirchhoff migration image, and $\mathbf{x}'$ is the trial image point.

---

[1] x_0 corresponds to the sign of p. For example, if $p > 0$, x_0 is the leftmost source position, and if $p < 0$, x_0 is the rightmost source position.

[2] Here, the plane-wave source field does not suffer from geometric spreading.

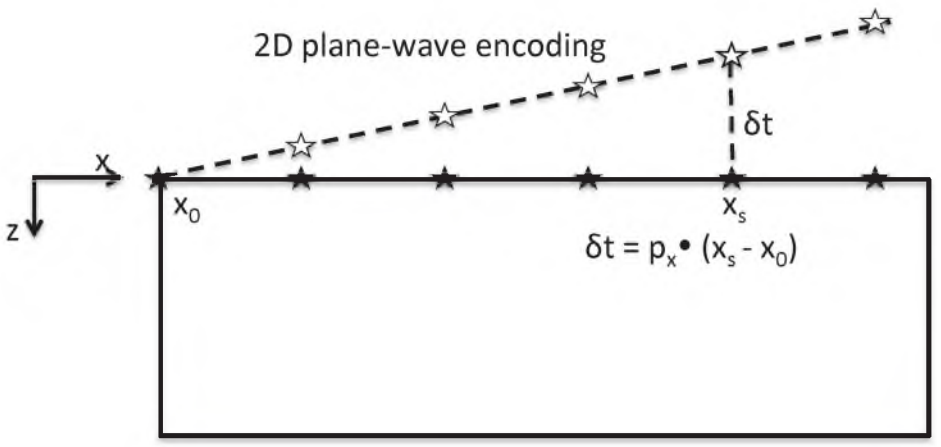

Figure 4.1. 2D plane-wave encoding, where the time shift is a linear function with respect to the source location x_s, reference source location x_0 and the slope is the ray parameter p.

Equations 4.3 and 4.4 can be written in the matrix-vector notation:

$$\widehat{\mathbf{d}}_p = \widehat{\mathbf{L}}_p \mathbf{m}, \tag{4.5}$$

$$\widehat{\mathbf{m}} = \widehat{\mathbf{L}}_p^{\dagger} \widehat{\mathbf{d}}_p, \tag{4.6}$$

and for N_P plane-waves, we have

$$\widehat{\mathbf{d}} = \begin{bmatrix} \widehat{\mathbf{d}}_1 \\ \widehat{\mathbf{d}}_2 \\ \vdots \\ \widehat{\mathbf{d}}_{N_P} \end{bmatrix} = \begin{bmatrix} \widehat{\mathbf{L}}_1 \\ \widehat{\mathbf{L}}_2 \\ \vdots \\ \widehat{\mathbf{L}}_{N_P} \end{bmatrix} \mathbf{m} = \sum_{i=1}^{N_P} \widehat{\mathbf{L}}_i \mathbf{m} = \widehat{\mathbf{L}} \mathbf{m}, \tag{4.7}$$

$$\widehat{\mathbf{m}} = \begin{bmatrix} \widehat{\mathbf{L}}_1^{\dagger} & \widehat{\mathbf{L}}_2^{\dagger} & \cdots & \widehat{\mathbf{L}}_{N_P}^{\dagger} \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{d}}_1 \\ \widehat{\mathbf{d}}_2 \\ \vdots \\ \widehat{\mathbf{d}}_{N_P} \end{bmatrix} = \sum_{i=1}^{N_P} \widehat{\mathbf{L}}_i^{\dagger} \widehat{\mathbf{d}}_i = \widehat{\mathbf{L}}^{\dagger} \widehat{\mathbf{d}}. \tag{4.8}$$

For multisource phase-encoding, single shot gathers are shifted with random time delays or encoded with random polarities of ± 1 or both, so crosstalk noise is introduced into the migration image. However, for plane-wave encoding, single shot gathers are shifted with a linear function, so that the plane-wave gathers are coherent signals, and there is no crosstalk noise. Instead, it contains aliasing artifacts (as shown in Figure 4.2a), which can be reduced by stacking images from different angles (as shown in Figure 4.2b). Stork and Kapoor (2004), Etgen (2005), Zhang et al. (2005), and Vigh and Starr (2008) provided estimates on the number of angles needed to provide an un-aliased stacked image. Their analysis showed that the number of angles (p-values) depends on the recording aperture, the velocity model, the source bandwidth and the maximum dip angle to be imaged. Figure 4.2c shows that only 11 angles are not sufficient to image the subsurface structure of Marmousi2 model, but 51 angles can provide a good quality image.

4.2.3 Plane-wave Least-squares Migration

Similar to shot-domain LSM, a misfit function for the plane-wave data can be expressed as:

$$f(\widehat{\mathbf{m}}) = \frac{1}{2} \| \widehat{\mathbf{L}} \widehat{\mathbf{m}} - \widehat{\mathbf{d}} \|_2^2, \tag{4.9}$$

and the plane-wave least-squares Kirchhoff migration (PWLSM) can be defined as searching for a reflectivity model $\widehat{\mathbf{m}}$ to minimize this function. By defining the observed data as $\widehat{\mathbf{d}}$, the plane-wave Kirchhoff modeling operator as $\widehat{\mathbf{L}}$ and the migration operator as $\widehat{\mathbf{L}}^{\dagger}$, a preconditioned conjugate gradient implementation can be expressed as:

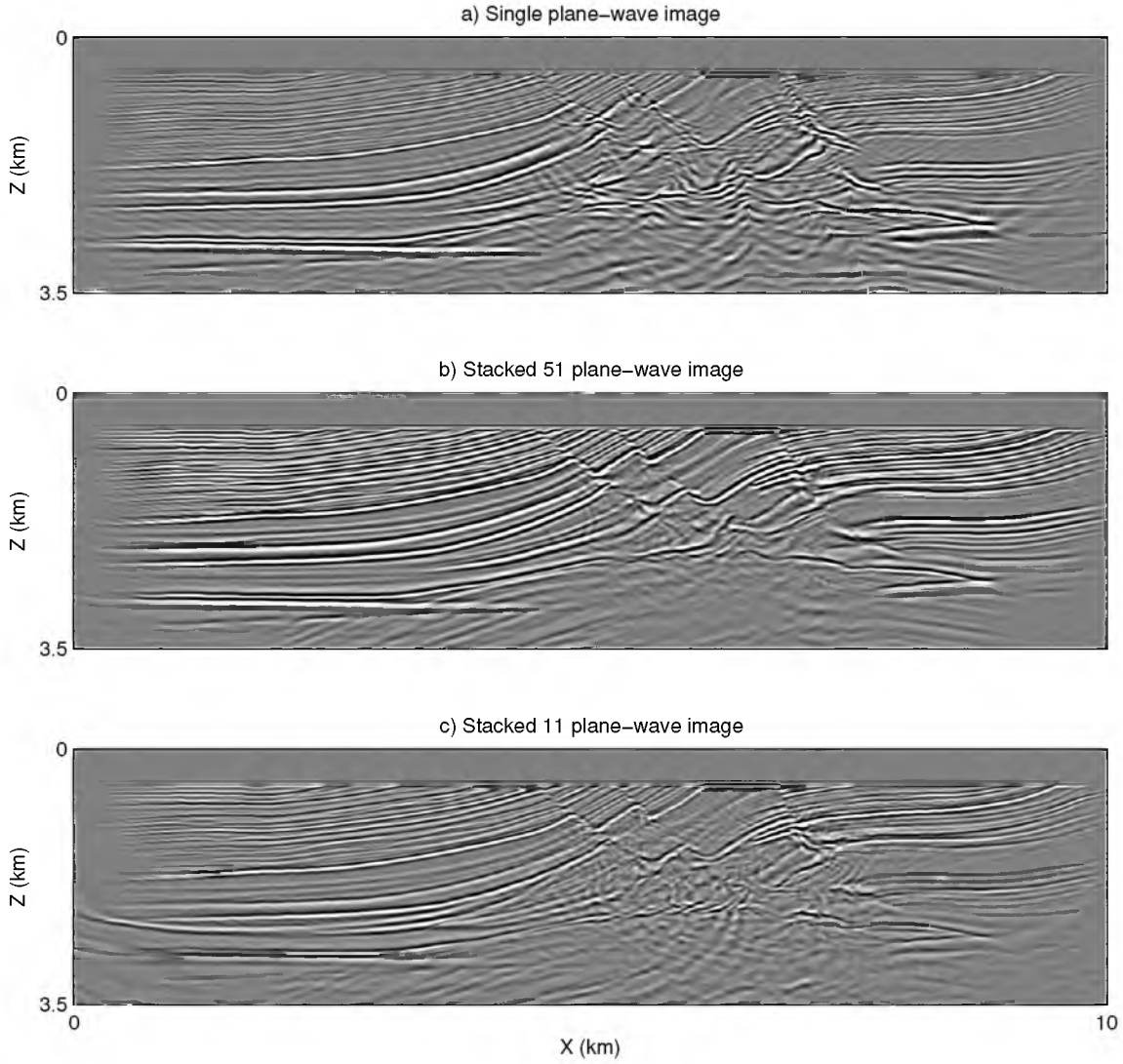

Figure 4.2. PWKM images on the Marmousi2 model: a) single plane-wave image, b) and c) are the stacked images computed from 51 and 11 plane-waves images, respectively.

1. calculate the data residual at the $(k)^{th}$ iteration : $\delta\widehat{\mathbf{d}}^{(k)} = \widehat{\mathbf{L}}\widehat{\mathbf{m}}^{(k)} - \widehat{\mathbf{d}}$;
2. calculate the $(k+1)^{th}$ gradient: $\mathbf{g}^{(k+1)} = \widehat{\mathbf{L}}^{\dagger}\delta\mathbf{d}^{(k)}$;
3. calculate the conjugate direction: $\mathbf{z}^{(k+1)} = \mathbf{P}\mathbf{g}^{(k+1)} + \beta\mathbf{z}^{(\mathbf{k})}$, where $\mathbf{P}$ is a preconditioner and $\beta = \frac{\langle \mathbf{g}^{(k+1)}, \mathbf{P}\mathbf{g}^{(k+1)} \rangle}{\langle \mathbf{g}^{(k)}, \mathbf{P}\mathbf{g}^{(k)} \rangle}$;
4. calculate the step length: $\alpha = \frac{\langle \mathbf{z}^{(k+1)}, \mathbf{g}^{k+1} \rangle}{\langle \widehat{\mathbf{L}}\mathbf{z}^{(k+1)}, \widehat{\mathbf{L}}\mathbf{z}^{(k+1)} \rangle}$;
5. update the model: $\widehat{\mathbf{m}}^{(k+1)} = \widehat{\mathbf{m}}^{(k)} - \alpha\mathbf{z}^{(k+1)}$;
6. repeat steps 1 - 5 until an acceptable data residual is achieved or the iteration reaches the user-defined maximum iteration number.

4.2.4 Regularize Plane-wave Least-squares Migration

If the migration velocity is not accurate, the prestack images from different plane-wave gathers are inconsistent with one another. Therefore simple stacking might blur the image and slow the convergence rate (Dai and Schuster, 2013; Wang et al., 2013).

In order to improve the robustness of plane-wave LSM in the presence of migration velocity errors, we assume that each plane-wave gather $\widehat{\mathbf{d}}_i$ is associated with its own migration image $\widehat{\mathbf{m}}_i$, so an ensemble of the prestack images $\widehat{\underline{\mathbf{m}}}$ can be defined as

$$\widehat{\underline{\mathbf{m}}} = \begin{bmatrix} \widehat{\mathbf{m}}_1 \\ \widehat{\mathbf{m}}_2 \\ \vdots \\ \widehat{\mathbf{m}}_{N_F} \end{bmatrix}, \tag{4.10}$$

and the modeling and migration equations can be expressed as

$$\widehat{\mathbf{d}} = \begin{bmatrix} \widehat{\mathbf{d}}_1 \\ \widehat{\mathbf{d}}_2 \\ \vdots \\ \widehat{\mathbf{d}}_{N_F} \end{bmatrix} = \begin{bmatrix} \widehat{\mathbf{L}}_1 & & & \\ & \widehat{\mathbf{L}}_2 & & \\ & & \vdots & \\ & & & \widehat{\mathbf{L}}_{N_F} \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{m}}_1 \\ \widehat{\mathbf{m}}_2 \\ \vdots \\ \widehat{\mathbf{m}}_{N_F} \end{bmatrix} = \widehat{\mathcal{L}}\ \widehat{\underline{\mathbf{m}}}, \tag{4.11}$$

and

$$\widehat{\underline{\mathbf{m}}} = \begin{bmatrix} \widehat{\mathbf{m}}_1 \\ \widehat{\mathbf{m}}_2 \\ \vdots \\ \widehat{\mathbf{m}}_{N_F} \end{bmatrix} = \begin{bmatrix} \widehat{\mathbf{L}}_1^{\dagger} & & & \\ & \widehat{\mathbf{L}}_2^{\dagger} & & \\ & & \vdots & \\ & & & \widehat{\mathbf{L}}_{N_F}^{\dagger} \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{d}}_1 \\ \widehat{\mathbf{d}}_2 \\ \vdots \\ \widehat{\mathbf{d}}_{N_F} \end{bmatrix} = \widehat{\mathcal{L}}^{\dagger}\ \widehat{\mathbf{d}}. \tag{4.12}$$

A new misfit function is defined as

$$f(\widehat{\underline{\mathbf{m}}}) = \frac{1}{2}\|\widehat{\mathcal{L}}\widehat{\underline{\mathbf{m}}} - \widehat{\mathbf{d}}\|_2^2 + \mathbf{R}. \tag{4.13}$$

The regularization term $\mathbf{R}$ is defined as a function that penalizes the difference between migration images computed with slightly different incidence angles:

$$\mathbf{R} = \frac{1}{2}\gamma \sum_{i=1}^{N_P-1} ||\widehat{\mathbf{m}}_{i+1} - \widehat{\mathbf{m}}_i||_2^2, \tag{4.14}$$

$$= \frac{1}{2}\gamma || \begin{bmatrix} -1 & 1 & & & & \\ & -1 & 1 & & & \\ & & \vdots & \vdots & & \\ & & & -1 & 1 & \\ & & & & -1 & 1 \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{m}}_1 \\ \widehat{\mathbf{m}}_2 \\ \vdots \\ \widehat{\mathbf{m}}_{N_P-1} \\ \widehat{\mathbf{m}}_{N_P} \end{bmatrix} ||_2^2, \tag{4.15}$$

$$= \frac{1}{2}\gamma ||C\widehat{\underline{\mathbf{m}}}||_2^2, \tag{4.16}$$

where γ is the damping coefficient determined by trial-and-error testing. We denote this method as regularized plane-wave least-squares migration (RPWLSM).

The preconditioned conjugate gradient implementation is similar to PWLSM, except that:

1. the modeling and migration operators $\widehat{\mathbf{L}}$ and $\widehat{\mathbf{L}}^\dagger$ are replaced with the new operators $\widehat{\mathcal{L}}$ and $\widehat{\mathcal{L}}^\dagger$,
2. the stacked image $\widehat{\mathbf{m}}$ is replaced with the ensemble of the prestack plane-wave images $\widehat{\underline{\mathbf{m}}}$,
3. the gradient $\mathbf{g}^{(k+1)}$ includes the partial derivative of $\frac{\partial \mathbf{R}}{\partial \widehat{\underline{\mathbf{m}}}}$, such that $\mathbf{g}^{(k+1)} = \widehat{\mathcal{L}}^\dagger \delta\widehat{\mathbf{d}}^{(k)} + \gamma C^T C\widehat{\underline{\mathbf{m}}}$;
4. the analytical step length α includes the modified term: $\alpha = \frac{\langle \mathbf{z}^{(k+1)}, \mathbf{g}^{k+1} \rangle}{\langle \widehat{\mathbf{L}}\mathbf{z}^{(k+1)}, \widehat{\mathbf{L}}\mathbf{z}^{(k+1)} \rangle + \gamma ||C\mathbf{z}^{(k+1)}||_2^2}$.

4.2.5 Plane-wave Eikonal Solver

4.2.5.1 Theory

A key step in Kirchhoff migration is the travel time calculation. Traditional methods of travel time calculation fall into two main groups: ray tracing methods (Julian and Gubbins, 1977; Virieux and Farra, 1991; Červený, 2005) and the eikonal equation solvers (Vidale, 1988; van Trier and Symes, 1991; Qin et al., 1992; Sethian and Popovici, 1999; Zhao, 2005). Vidale (1988) introduced a finite-difference (FD) solver to the eikonal equation along an expanding-square. However, this approach encounters difficulties with high contrast velocity interfaces. Qin et al. (1992) developed an expanding wavefront FD solver to remedy this problem and accurately calculate the first-arrival travel times. In this chapter, I introduce

a FD solver to the 2D eikonal equation to calculate the plane-wave travel times based on the expanding wavefront FD scheme (Qin et al., 1992).

The 2D eikonal equation can be expressed as

$$(\frac{\partial T}{\partial x})^2 + (\frac{\partial T}{\partial z})^2 = s(x,z)^2, \tag{4.17}$$

where $T(x,z)$ is the arrival time for seismic energy through a medium with the slowness $s(x,z)$. The numerical procedures for the expanding wavefront method for a point source are described below:

1. calculate the travel times at grids around the source point (x_s, z_s) within a user-defined $N \times N$ square of gridpoints,
2. define the wavefront with an outer perimeter of timed grids,
3. sort the wavefront and find the *global minimum* with the least travel time,
4. calculate the travel times for the unknown neighboring[3] gridpoints of the *global minimum*,
5. repeat steps 2-4 until all the untimed points are fulfilled.

The above FD eikonal solver is an efficient numerical algorithm for solving the nonlinear partial differential equation (PDE) with the stationary boundary value (Zhao, 2005)

$$T(x_s, z_s) = 0. \tag{4.18}$$

Plane-waves can be produced by placing a series of point sources with linear delayed shooting times. The boundary values of the plane-wave eikonal equation can be defined as

$$\sum_{i=1}^{N_x} T(x_i, 0) = p \cdot (x_i - x_0), \tag{4.19}$$

where p is the ray parameter, and x_0 is the x-coordinate corresponding to the source with the greatest time delay. In this section, I calculate the plane-wave travel times by applying the expanding wavefront eikonal solver with the boundary values in equation 4.19.

Figure 4.3 illustrates the expanding wavefront FD solver scheme. All the filled circles have been timed, while all the empty ones are untimed. The wavefront is formed from both red- and green-filled-circles, and the empty circles with green squares will be expanded. Figure 4.3a and b show two wavefronts. Figure 4.3c-e shows the three finite-difference stencils for travel times corresponding to the following three cases:

[3]The neighboring points are defined as those gridpoints that are immediately next to the evaluation point.

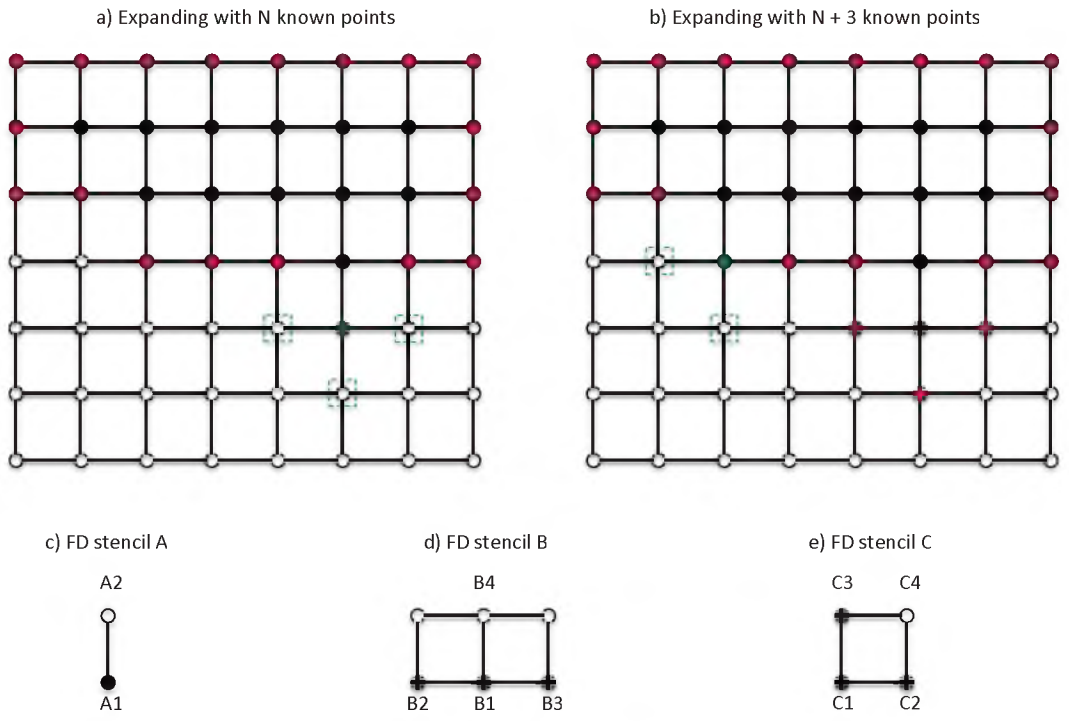

Figure 4.3. Gridpoint models for the 2D FD plane-wave eikonal solver: a) the current expanding wavefront with N known points, b) the next expanding wavefront with N+3 known points, c)-e) three different FD stencils. Green-filled-circles indicate the *global minimum* on the wavefront, red-filled-circles indicate all the other parts of the timed wavefront, and the black-empty-circles with green squares indicate the points that will be expanded.

1. the wavefront at time T_{A1} propagates until the time T_{A2}, where

$$T_{A2} = T_{A1} + h\bar{s}, \tag{4.20}$$
$$\bar{s} = \frac{1}{2}(s_{A1} + s_{A2}), \tag{4.21}$$

and h is the grid interval. This stencil will be used only when the *global minimum point* is on the edges of the model.

2. the wavefront at time T_{B1} propagates until the time T_{B4}, where

$$T_{B4} = \begin{cases} T_{B1} + \sqrt{\Delta}, & if \Delta \geq 0, \\ T_{B1} + h\bar{s}, & if \Delta < 0, \end{cases} \tag{4.22}$$
$$\Delta = h^2\bar{s}^2 - \frac{1}{4}(T_{B2} - T_{B3})^2, \tag{4.23}$$
$$\bar{s} = \frac{1}{2}(s_{B4} + \frac{1}{3}(s_{B1} + s_{B2} + + s_{B3})), \tag{4.24}$$

3. the wavefront at time T_{C1} propagates until the time T_{C4}, where

$$T_{C4} = \begin{cases} T_{C1} + \sqrt{\Delta}, & if \Delta \geq 0, \\ T_{C1} + \sqrt{2}h\bar{s}, & if \Delta < 0, \end{cases} \tag{4.25}$$
$$\Delta = 2h^2\bar{s}^2 - (T_{C2} - T_{C3})^2, \tag{4.26}$$
$$\bar{s} = \frac{1}{4}(s_{C1} + s_{C2} + s_{C3} + s_{C4}). \tag{4.27}$$

The accuracy of the FD stencils B and C is analyzed in Vidale (1988).

4.2.5.2 Numerical Tests

The proposed 2D plane-wave eikonal solver is first tested with a homogeneous velocity model. Figure 4.4a-d shows the plane-wave travel time contours with ray parameters 0.4, 0.2, 0 and -0.1, respectively. To examine the accuracy of the proposed method, a FD solution to the acoustic wave equation with linear delay-shots at p of 0.2 ms/m is simulated to produce six snapshots as shown in Figure 4.5a-f. Contours (red dots) with the associated travel times are superimposed on the snapshots in Figure 4.6. The coincidence indicates the accuracy of the method. The plane-wave eikonal solver is also tested on the Marmousi2 model in Figure 4.7. Similar to the homogeneous velocity test, four plane-wave travel time contours are shown in Figure 4.8a-d. The coincidence between the wavefronts in Figure 4.9 and the contours in Figure 4.10 illustrates the accuracy of this approach.

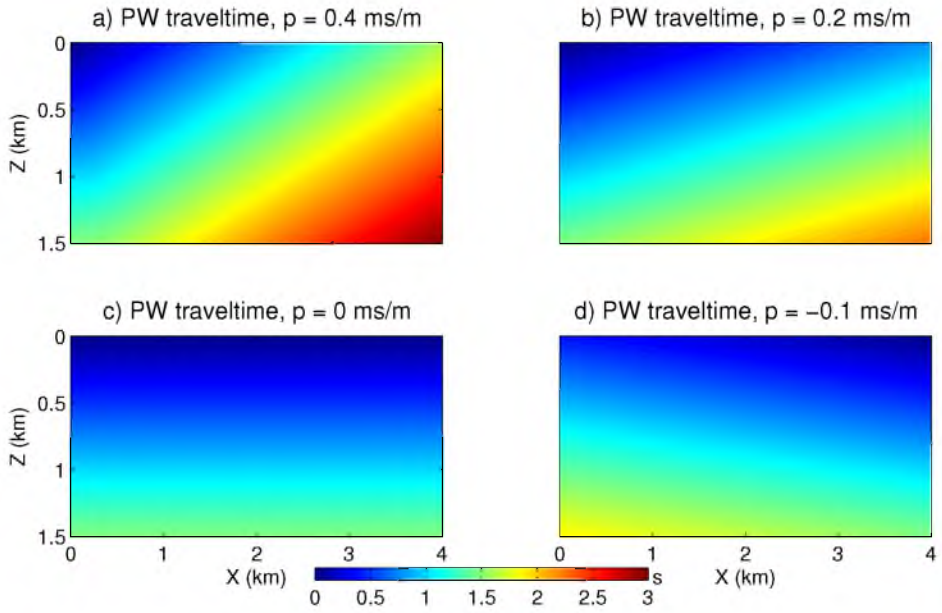

Figure 4.4. The four plane-wave travel time contours associated with the ray parameters 0.4, 0.2, 0 and -0.1 ms/m, respectively.

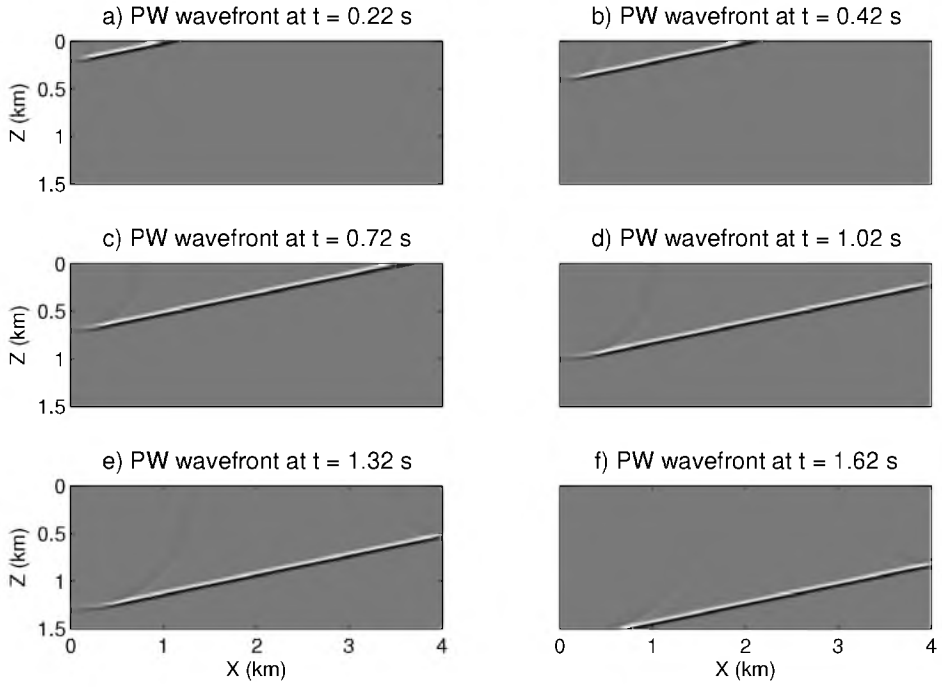

Figure 4.5. The six plane-wave snapshots at t 0.22, 0.42, 0.72, 1.02, 1.32 and 1.62 s, respectively. Snapshots are computed by a FD solution to the acoustic wave equation with plane-wave source of $p = 0.2$ ms/m.

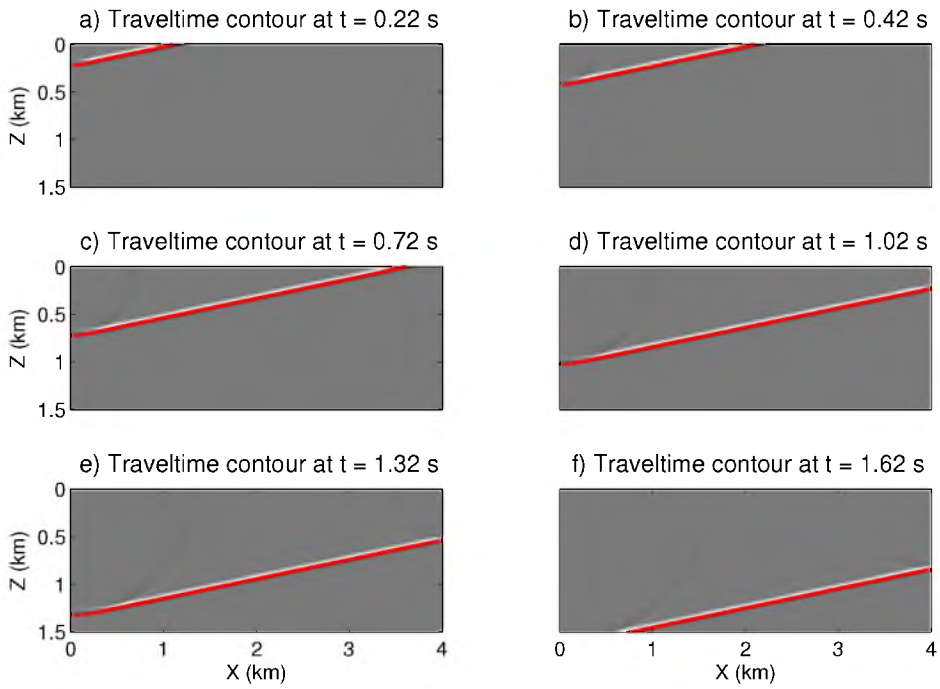

Figure 4.6. The six contours of the eikonal travel times superimposed on the snapshots computed by a FD solution to the wave equation.

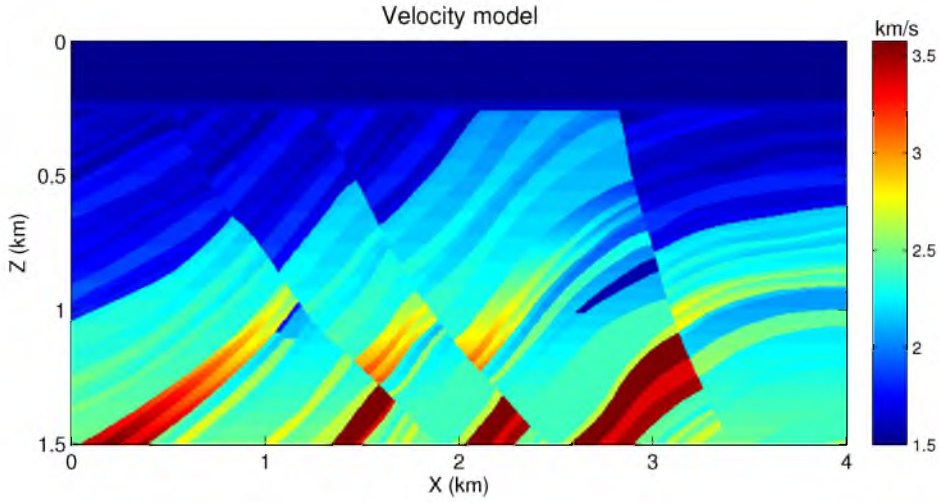

Figure 4.7. Marmousi2 model used for testing the plane-wave eikonal solver.

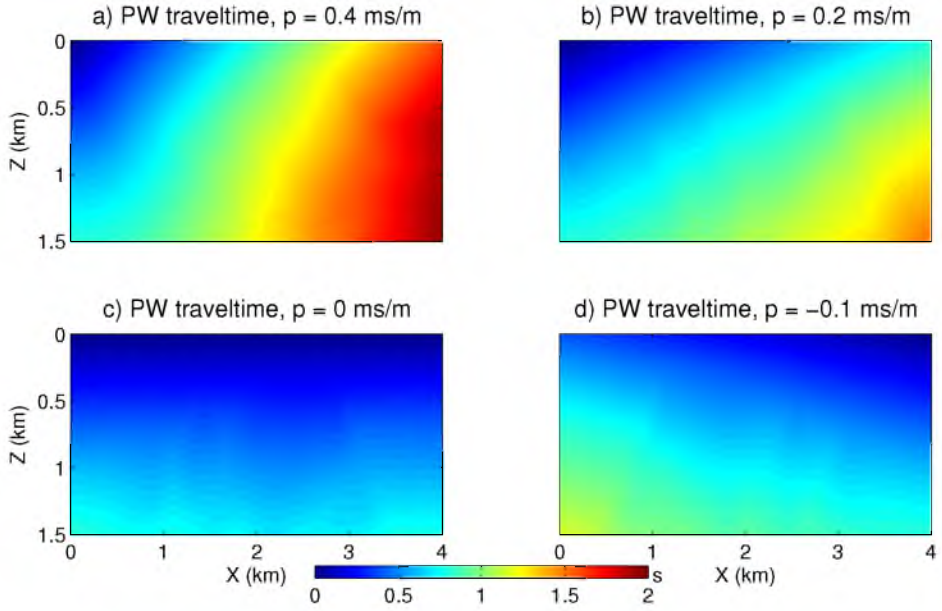

Figure 4.8. The four plane-wave travel time contours associated with the ray parameters 0.4, 0.2, 0 and -0.1 ms/m, respectively.

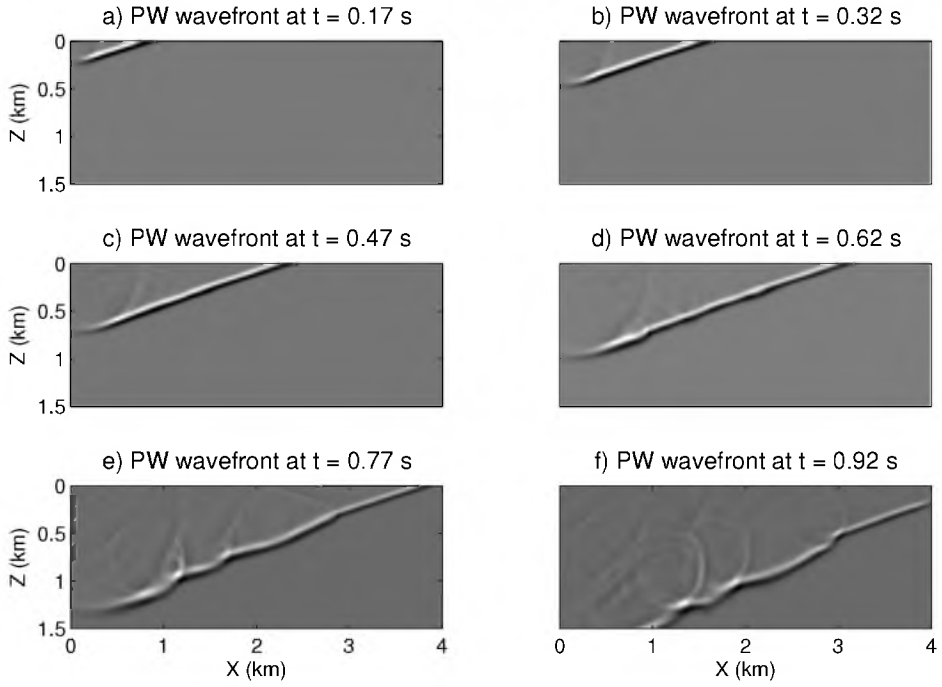

Figure 4.9. The six plane-wave snapshots at t 0.17, 0.32, 0.47, 0.62, 0.77 and 0.92 s, respectively. Snapshots are computed by a FD solution to the acoustic wave equation with plane-wave source of $p = 0.2$ ms/m.

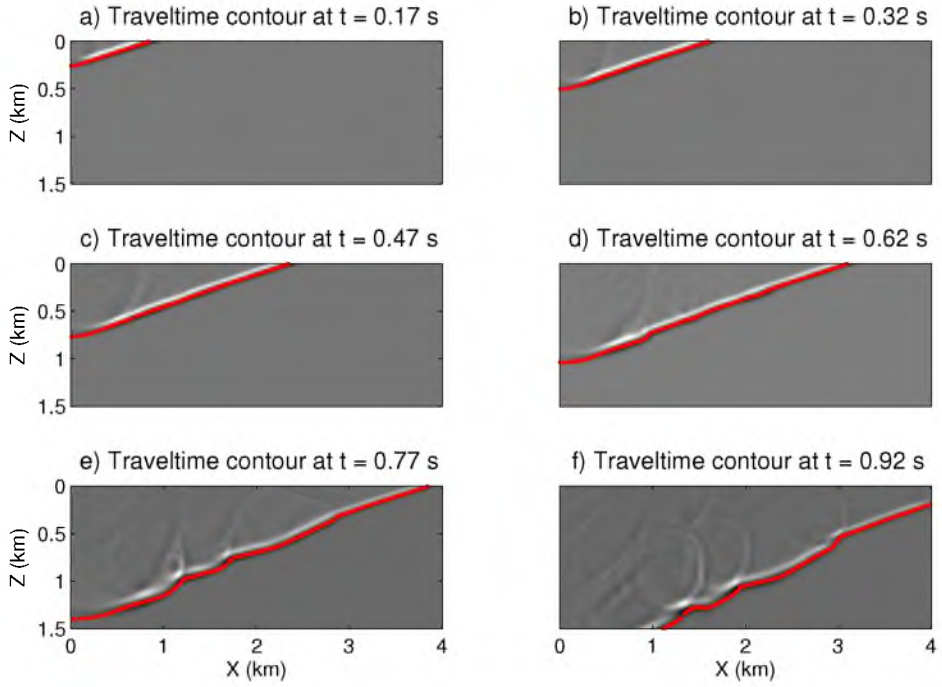

Figure 4.10. The six contours of the eikonal travel times superimposed on the snapshots computed by a FD solution to the wave equation.

4.3 Numerical Tests

The PWLSM and RPWLSM algorithms are tested on both synthetic and field data.

4.3.1 Synthetic Data Test

Synthetic data are generated for the 2D Marmousi2 model (Figure 4.11a) with Born modeling solution, where the model size is 1501 $\times$ 501 with a 5 m gridpoint interval. There are 301 shot gathers with a 25 m shot interval evenly distributed on the surface, and each shot gather shoots into 750 receivers with a 10 m receiver interval. A Ricker wavelet with a peak frequency of 15 Hz is used as the source wavelet. Figure 4.11b shows the smoothed migration velocity. To test the robustness of RPWLSM, an erroneous velocity model with a maximum 5% error (Figure 4.11c) is also tested. Forty-five plane-wave gathers are generated by the plane-wave encoding technique, with a range of ray parameters between -0.47 ms/m and 0.47 ms/m. Two plane-wave gathers are shown in Figure 4.12a-b with the ray parameters of -0.279 ms/m and 0.043 ms/m.

Figure 4.13a-b shows 301 common shot gathers (CSG) and 45 plane-wave gathers Kirchhoff migration images computed with an accurate velocity model. The PWKM image is almost identical to that of the CSG-domain KM image, which indicates that 45 plane-waves are sufficient for imaging the subsurface structure accurately. By replacing 301 CSGs with 45 plane-wave gathers, one can achieve a computational speed up of $\frac{301}{45} = 6.69$, and this efficiency can be further improved by decreasing the number of plane-waves. However, the image quality can be degraded by an insufficient number of plane-wave angles.

Figure 4.14a-b shows the PWLSM and the RPWLSM images after 10 iterations. Both of them are of better quality than the PWKM image in Figure 4.13a. It is also noted that when the migration velocity is accurate, PWLSM provides a high resolution image and has a faster convergence rate (Figure 4.15) than RPWLSM because the prestack images have reflectors that are aligned with one another. Hence, stacking plane-wave migration images together can suppress the migration artifacts.

The sensitivity of the migration images with respect to errors in Figure 4.11c-d is tested with the PWKM, PWLSM and RPWLSM methods. Figure 4.16a-c shows the PWKM, PWLSM and RPWLSM images computed with the erroneous migration velocity. The PWLSM image has the highest resolution at the shallow depth because there is no velocity error here, but for the deep part (circled with green dashed lines) RPWLSM provides the best image.

Common image gathers (CIG) from PWKM and RPWLSM are compared in Figure 4.17. Several areas marked with green dashed circles are selected to show the speedup advantages

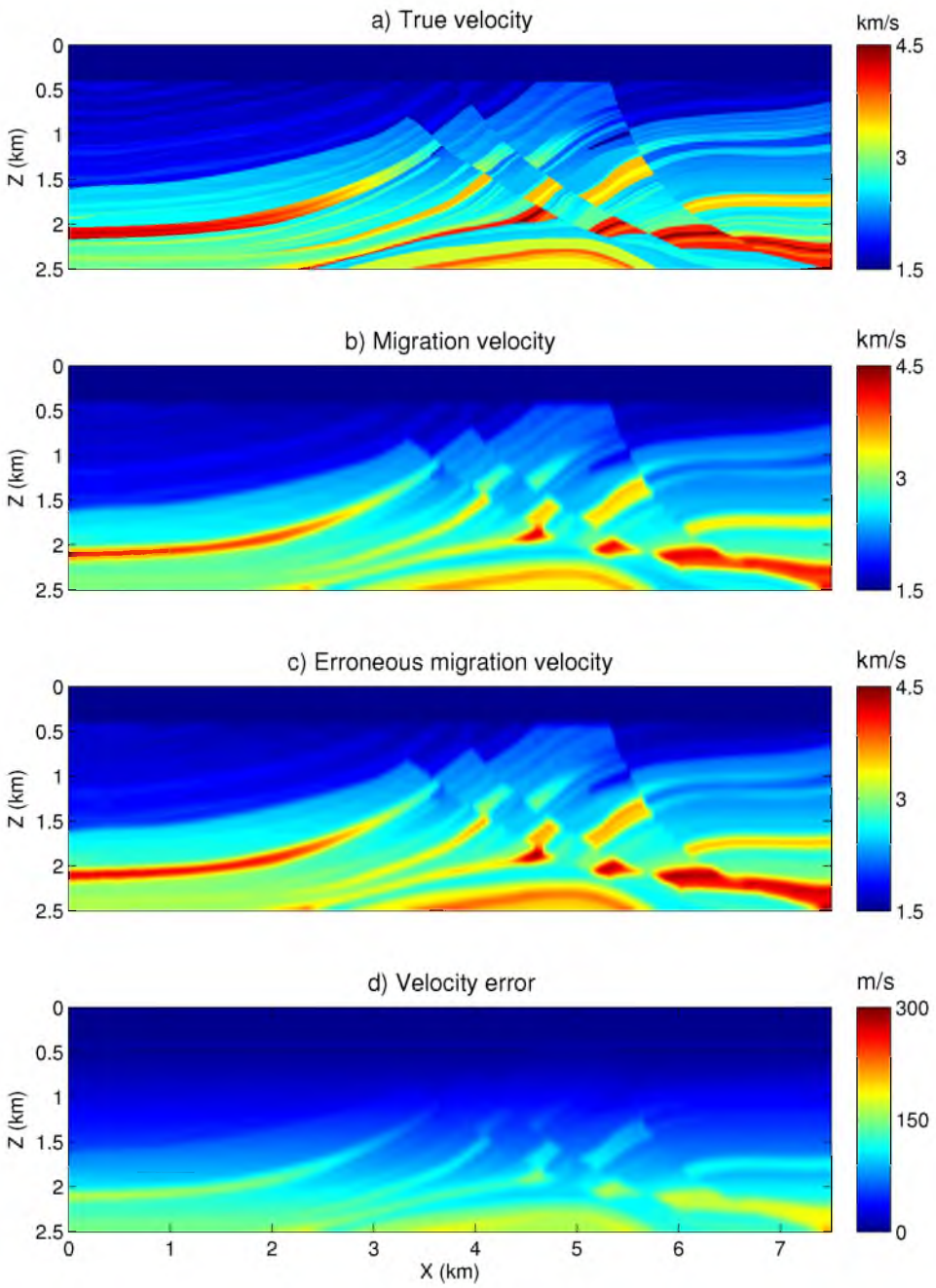

Figure 4.11. Marmousi2 velocity models: a) true velocity, b) smoothed migration velocity, c) erroneous velocity for sensitivity test, d) difference between b) and c). The velocity error will shift reflector positions by 2 and 2.5 wavelengths at the respective depths of 1.5 and 2 km.

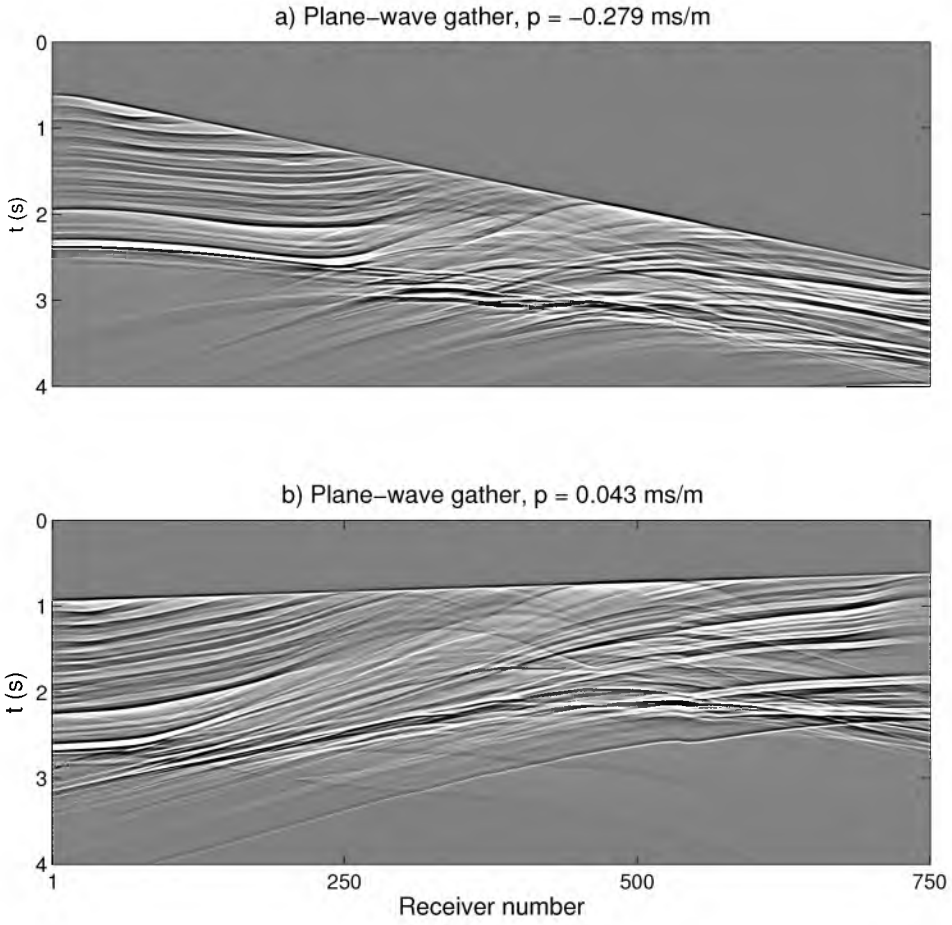

Figure 4.12. Plane-wave gathers with ray parameters of a) -0.279 and b) 0.043 ms/m.

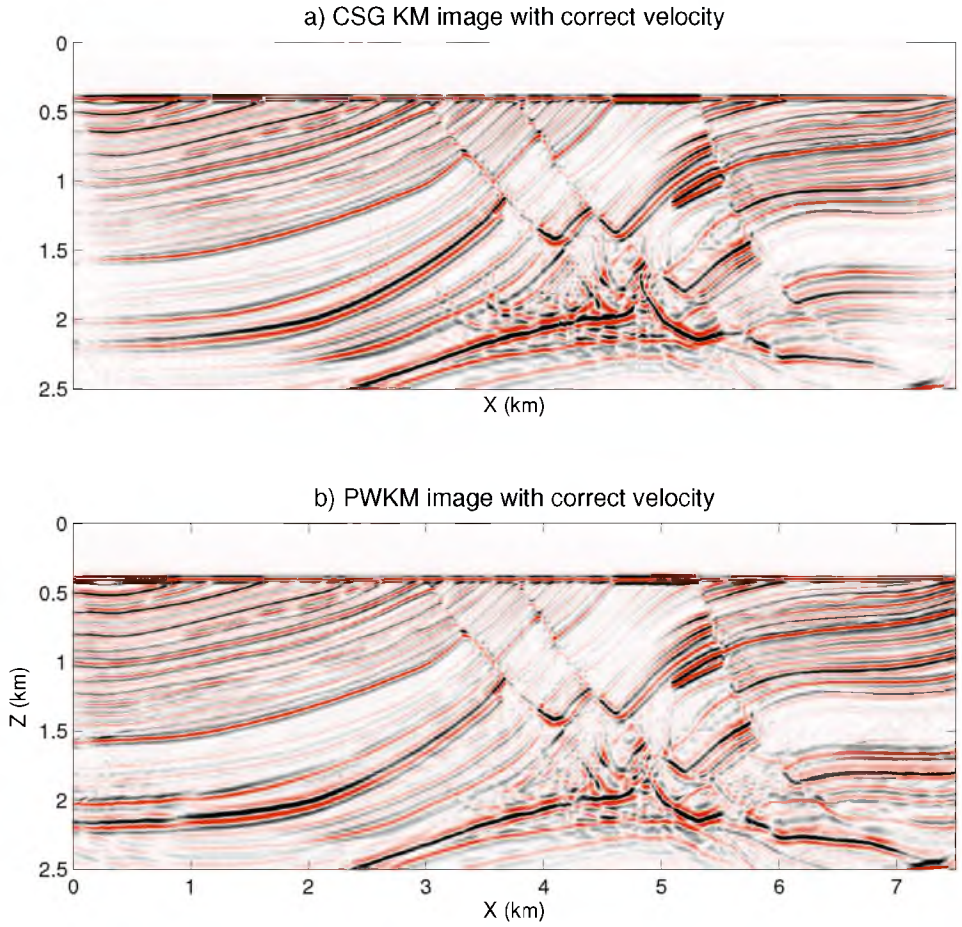

Figure 4.13. CSG KM and PWKM images computed with the true velocity model: a) CSM KM and b) PWKM images.

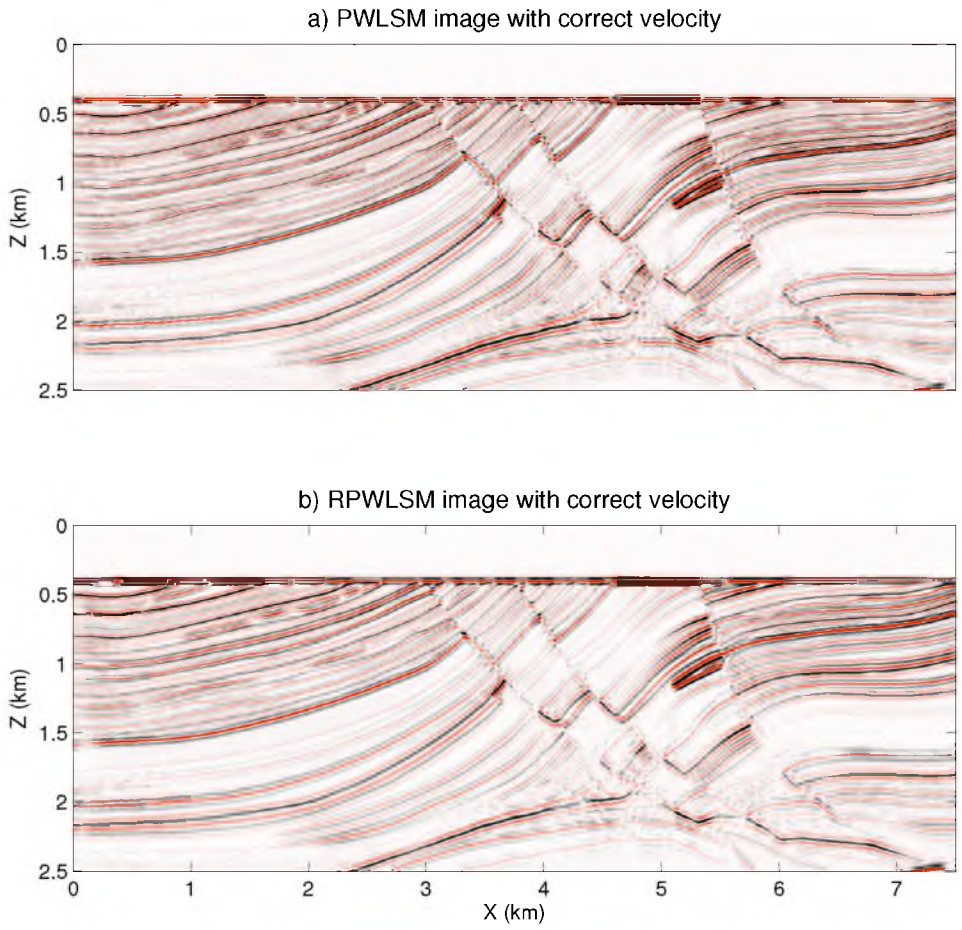

Figure 4.14. LSM images with accurate velocity model after 10 iterations: a) plane-wave least-squares migration and b) regularized plane-wave least-squares migration.

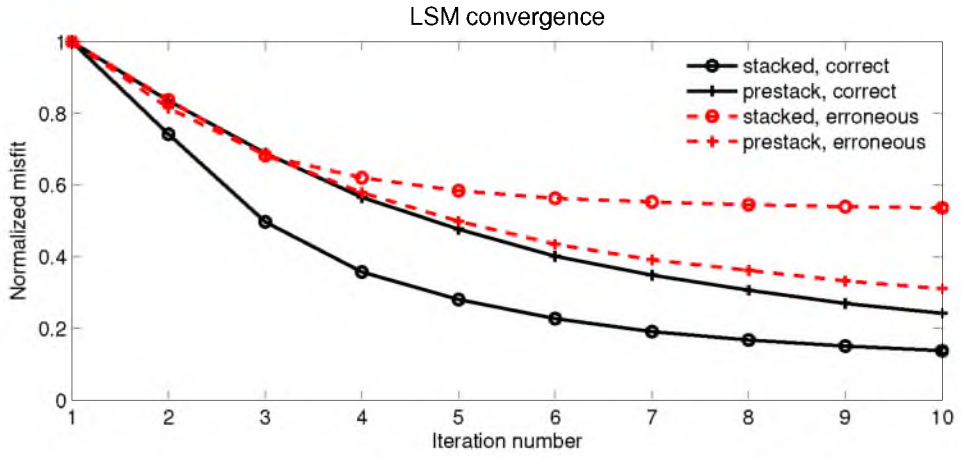

Figure 4.15. Convergent curves of PWLSM and RPWLSM for the true and erroneous migration velocities. Black lines are for the true migration velocity and red lines are the for the erroneous velocity model.

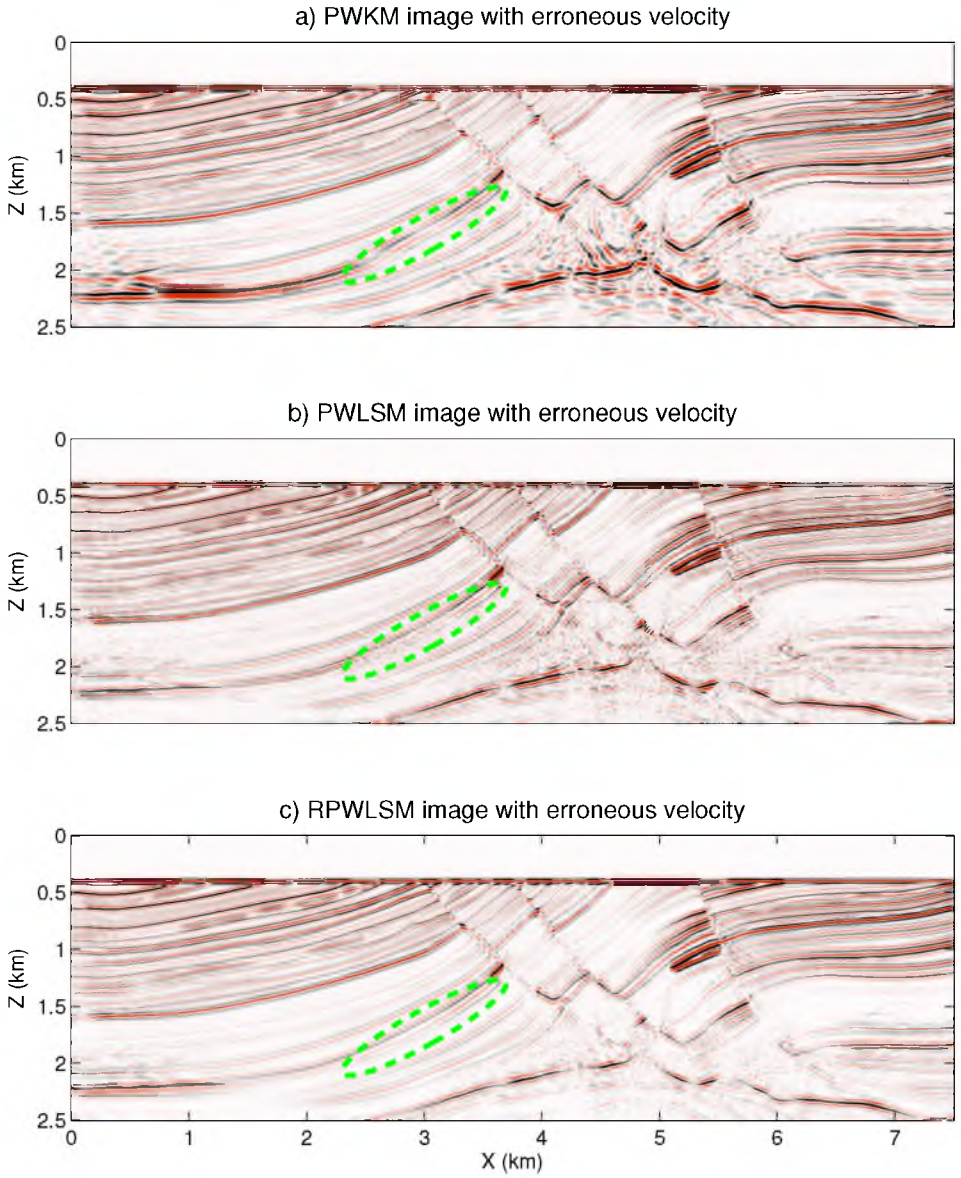

Figure 4.16. Migration images with erroneous velocity model for: a) PWKM, b) PWLSM and c) RPWLSM after 10 iterations.

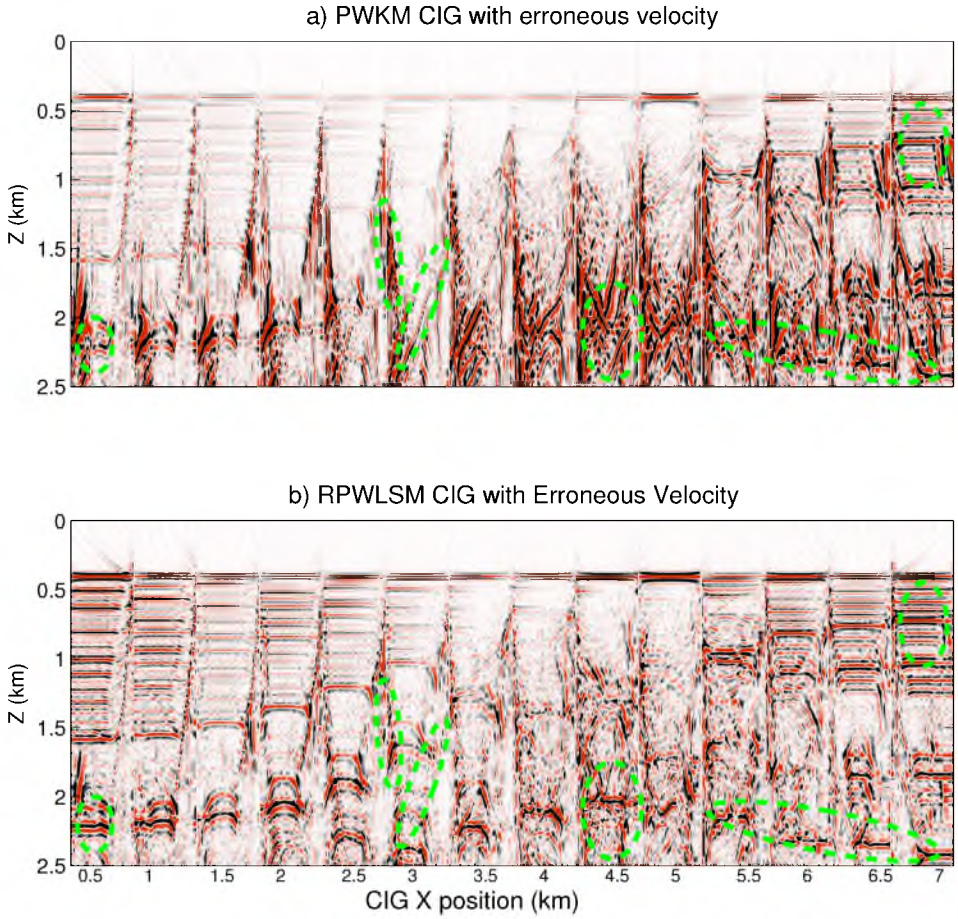

Figure 4.17. CIGs with erroneous velocity of: (a) PWKM and (b) RPWLSM after 10 iterations.

of RPWLSM.

Figure 4.15 compares the convergence rates of PWLSM and RPWLSM with both correct and erroneous velocity models. PWLSM converges faster than RPWLSM when the velocity model is accurate, while RPWLSM enjoys a faster convergence rate than PWLSM if the velocity model has bulk errors.

4.3.2 Field Data Test

The proposed methods are tested on a 2D marine data set. There are 515 shots with a shot interval of 37.5 m, and each shot is recorded by a 6 km long cable with 480 receivers with a 12.5 m receiver interval. The nearest receiver offset from the source is 198 m. The CSGs are first transformed into common midpoint profiles (CMP), then a normal moveout time shift is applied, followed by a 2D interpolation. The interpolated data are then transformed into common receiver gathers (CRG) with a split-spread acquisition geometry using reciprocity. A tau-p transform is applied to each CRG to generate 31 plane-wave gathers with ray parameters (p) ranging from -0.333 ms/m to 0.333 ms/m with an even sampling in p, and each plane-wave gather has 1260 traces. Figure 4.18a-b shows two CSGs, and Figure 4.18c-d shows two plane-wave gathers. The source wavelet is estimated by stacking the near-offset traces with a strong water-bottom reflection. Waveform inversion (Altheyab, 2012) is used to get the migration velocity model shown in Figure 4.19, and the model size is 2519×581 with a gridpoint interval of 6.25 m.

Figure 4.20a-b shows, respectively, the CSG KM and PWKM images. The image quality is comparable, and the shallow reflectors are highly resolved, while the deep part contains more artifacts. Figure 4.21a-b shows the PWLSM and RPWLSM images after 10 iterations, and a trim statics (see Appendix A) technique is applied to the prestack RPWLSM and produces the image (Figure 4.21c). Figures 4.22 and 4.23 show the two zoom views (marked as blue and red boxes) of images from PWKM, PWLSM, RPWLSM and RPWLSM with the trim statics. Similar to the synthetic results, Figure 4.22a-c shows that at the shallow depth both the PWLSM and RPWLSM images are of better quality than the PWKM image, but the PWLSM image has slightly higher quality than the RPWLSM image which indicates the velocity at shallow depth is accurate. However, for the deep part (as shown in Figure 4.23a-c, the PWLSM image shows a much higher level of noise, and the RPWLSM provides a better quality image with fewer artifacts and shows more continuous reflectors but with lower resolution. The image quality of PWKM is the best among the three, and reflectors are more continuous in Figure 4.23d after the trim statics technique (see Appendix A).

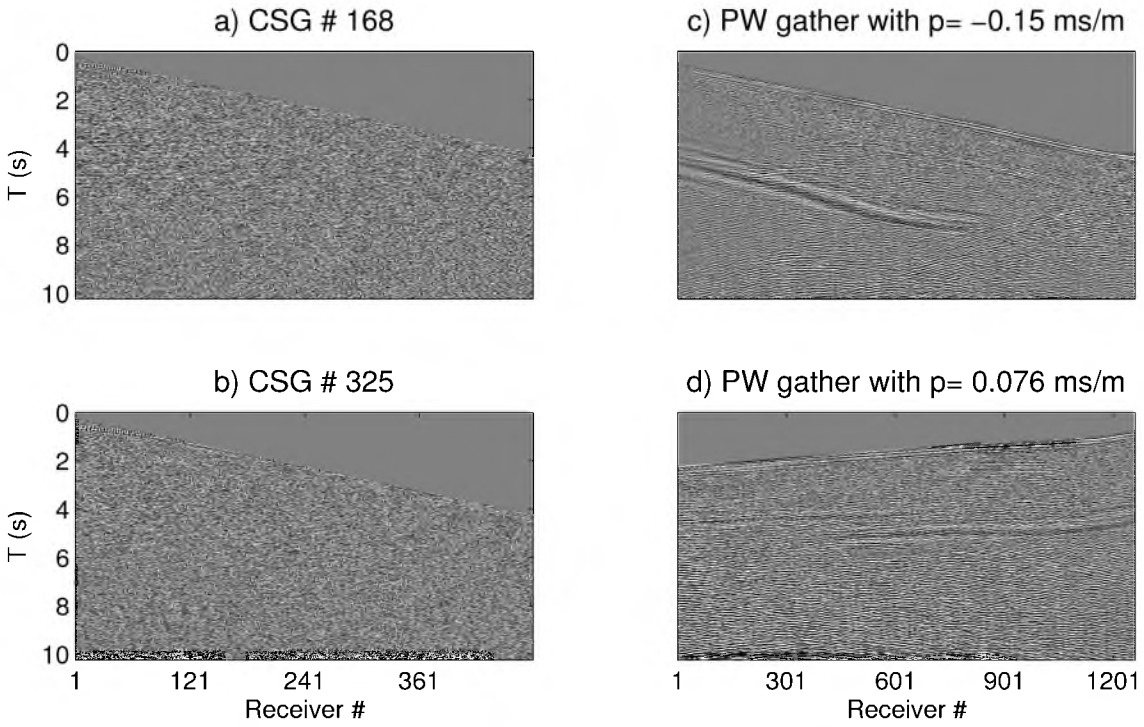

Figure 4.18. CSG and PW gathers: a)-b) two CSGs and c-d) two plane-wave gathers.

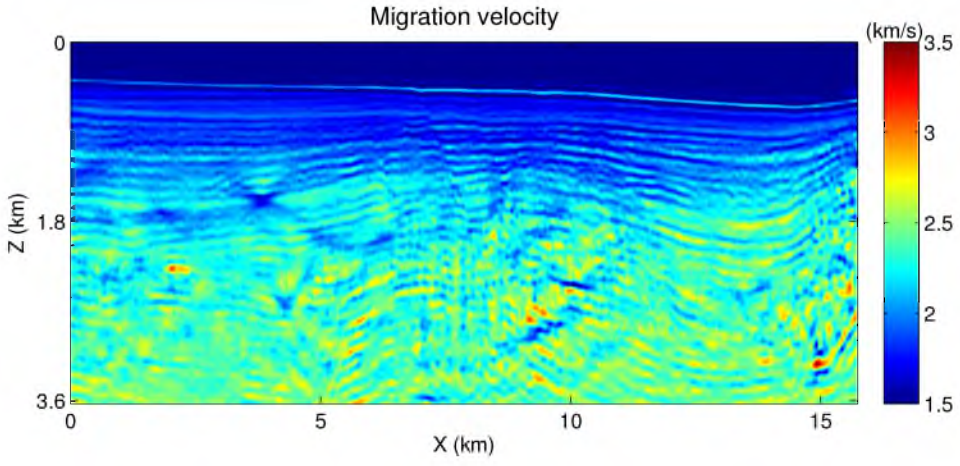

Figure 4.19. Migration velocity of the 2D Marine data.

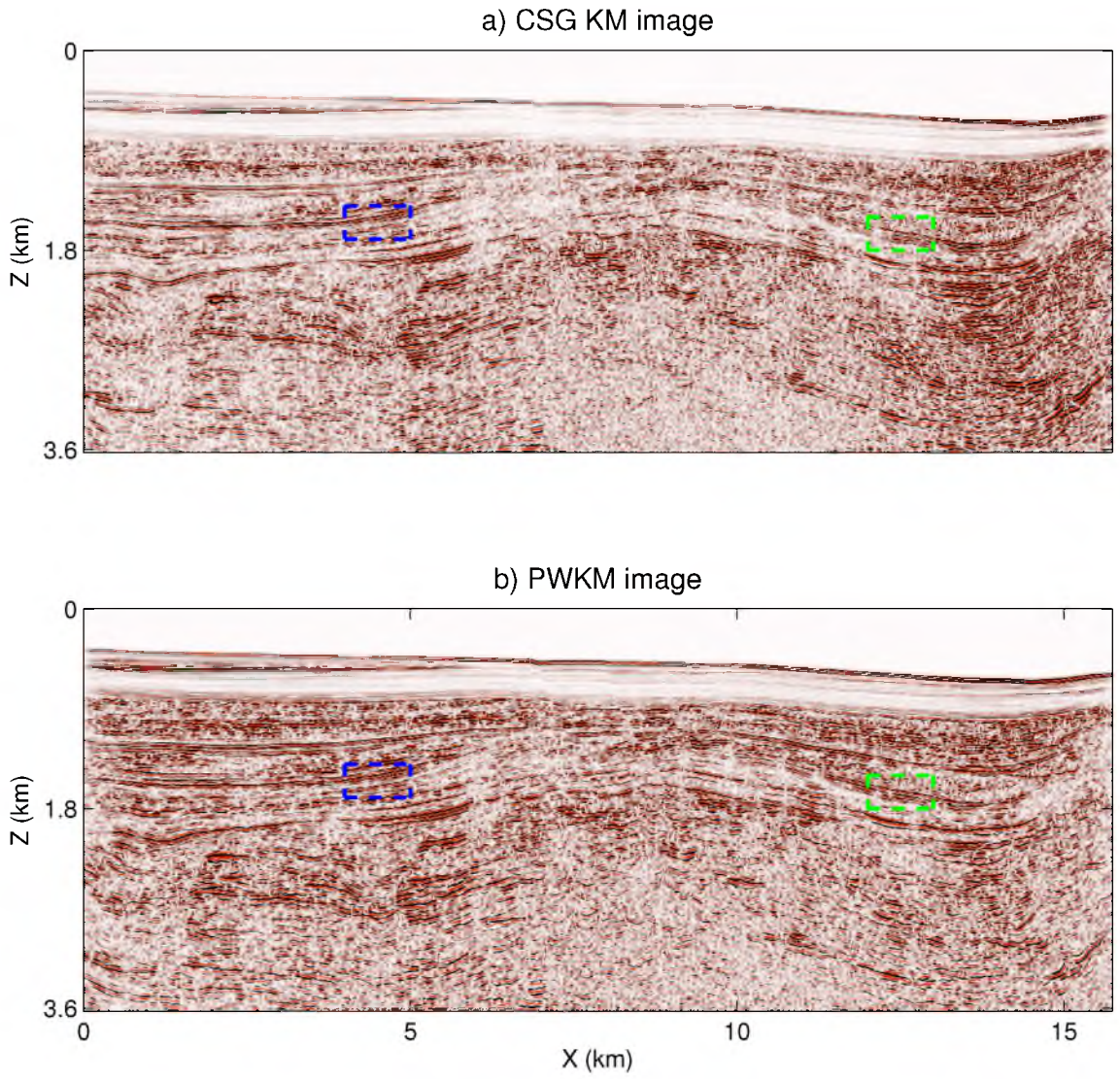

Figure 4.20. Images of: a) conventional KM and b) PWKM.

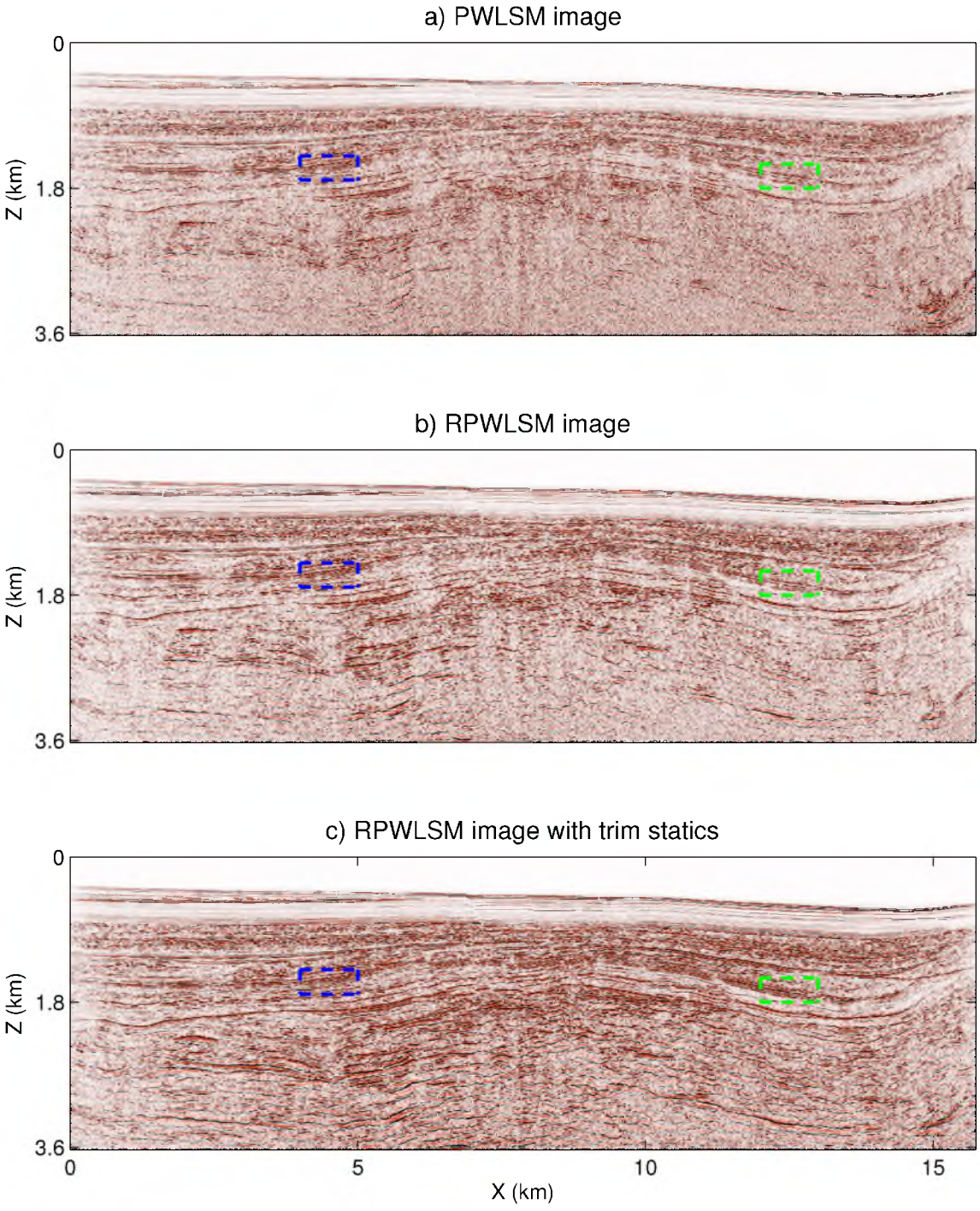

Figure 4.21. Images of: a) PWLSM, b) RPWLSM and c) RPWLSM after applying trim statics.

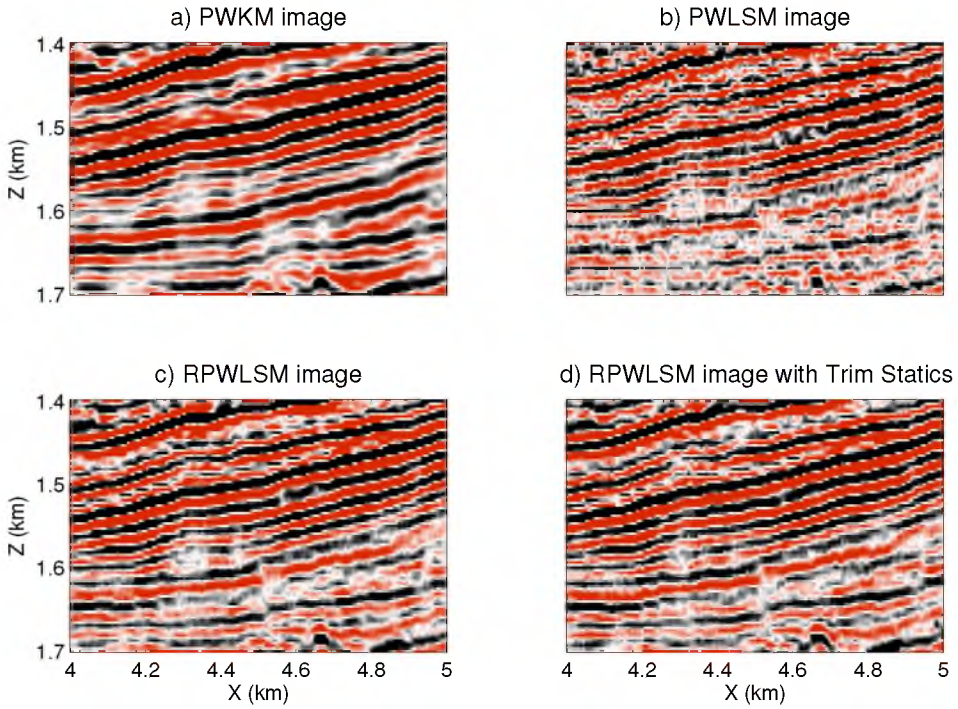

Figure 4.22. Zoom views of the blue boxes in Figures 4.20b and 4.21a-c.

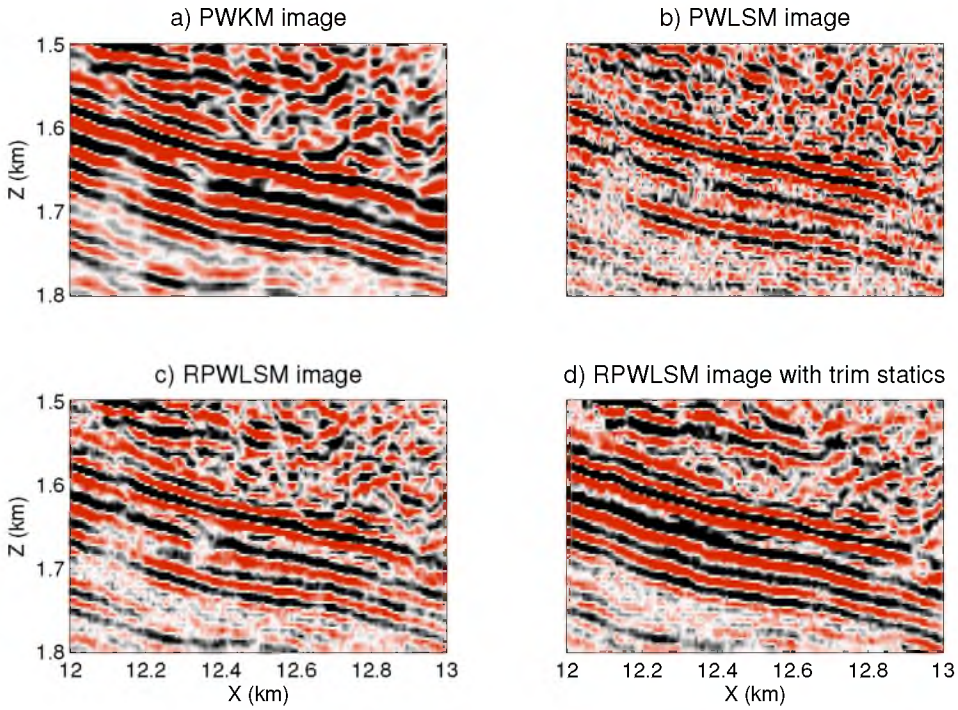

Figure 4.23. Zoom views of the red boxes in Figures 4.20b and 4.21a-c.

By analyzing the CIGs of PWKM and RPWLSM (Figure 4.24a-b), and the prestack PWKM and RPWLSM images with a zero incidence angle (Figure 4.25a-b), it is found that RPWLSM increases the prestack image resolution more than KM; but after stacking the final image is blurred which is a symptom of an inaccurate velocity model. Trim statics is the cure to this blurring due to inaccurate velocities. In addition, the high-resolution prestack images can be used to correct the volocity model by migration velocity analysis.

Figure 4.26 shows the plots of residual vs iteration number for these two LSM methods. PWLSM provides a reflectivity model that reduces the RMS misfit value to 55%, compared to 30% for RPWLSM. The computational and IO cost of RPWLSM and PWLSM are about $\frac{31\times1260}{515\times480} = 0.158$ of that for shot domain LSM.

4.4 Conclusions

A plane-wave least-squares migration algorithms is proposed to efficiently produce high quality images. By transforming CSG data into the plane-wave domain data, the computation and IO costs are significantly reduced. The success of PWLSM vanishes when the migration velocity model contains significant errors. To improve the robustness of this algorithm, a regularized plane-wave least-squares migration method is proposed and shown to give the most focused images in the presence of migration velocity errors. In addition, the prestack CIGs provide enhanced opportunities for velocity analysis. A 2D FD eikonal solver is introduced to calculate the plane-wave travel times for plane-wave Kirchhoff migration. Synthetic and field results show that: 1) PWLSM can provide the best quality images when the migration velocity is accurate, 2), the prestack RPWLSM is more robust when the velocity model contains a bulk of errors, 3), the high resolution of prestack images provided by RPWLSM vanishes by simple stacking. The remedy is a trim statics technique applied to the prestack gathers.

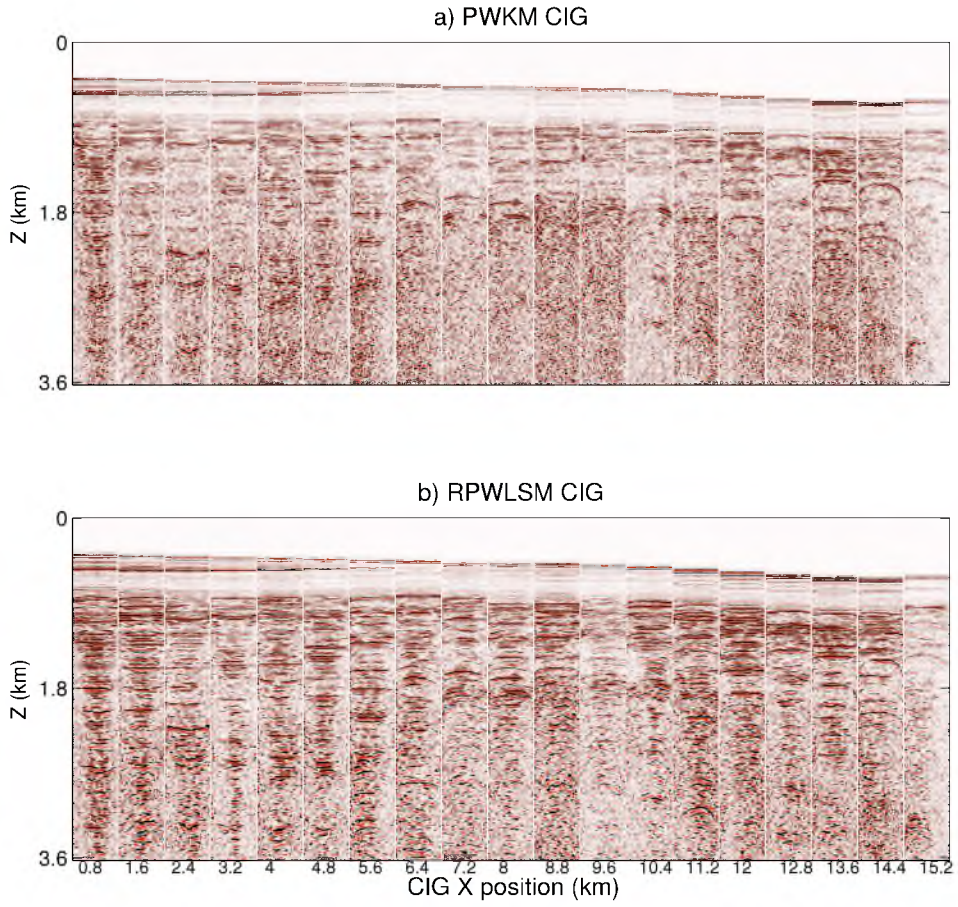

Figure 4.24. CIGs of: a) PWKM and b) RPWLSM.

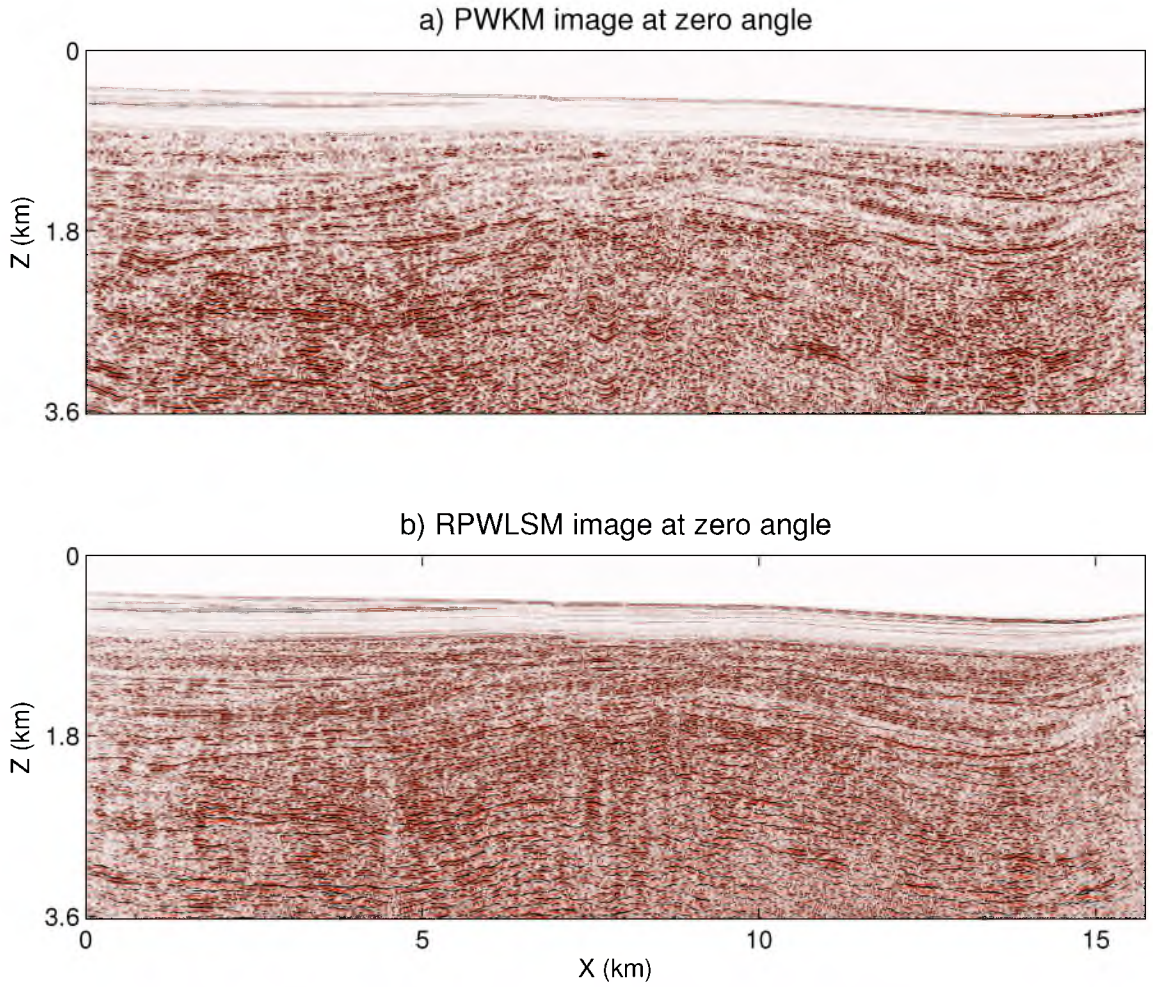

Figure 4.25. Images of: a) PWKM and b) RPWLSM with zero incident angle.

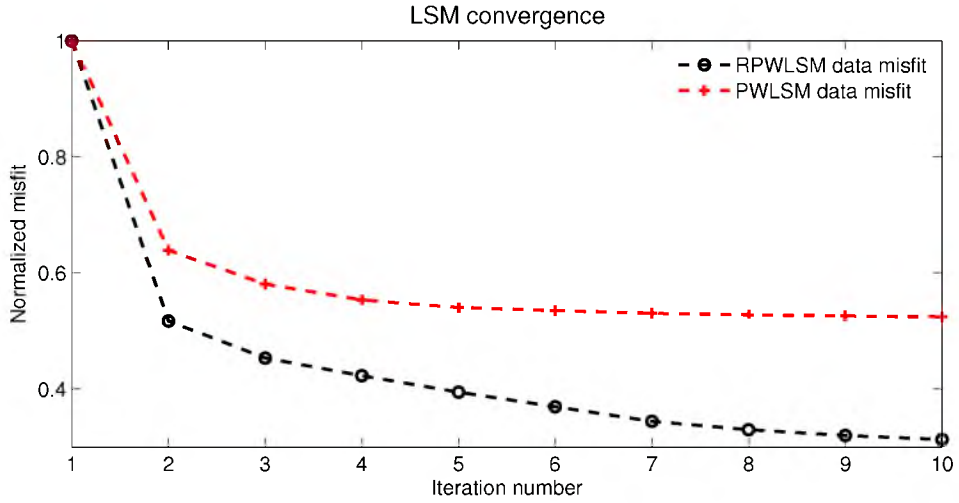

Figure 4.26. Convergent curves for PWLSM (red) and RPWLSM (black).

CHAPTER 5

3D PLANE-WAVE LEAST-SQUARES KIRCHHOFF MIGRATION

In this chapter, I generalize plane-wave Kirchhoff LSM from 2D to 3D. Instead of only one ray parameter, two ray parameters p_x and p_y in both the inline and the crossline directions are required for 3D plane-wave encoding. For the current 3D streamer acquisition geometry, the source sampling in the crossline direction is usually too sparse to support a full 3D plane-wave encoding, so an inline only plane-wave encoding is adopted and this is equivalent to a cylindrical-wave encoding (or line-source encoding for 3D geometry). The finite-difference solvers to the 3D eikonal equation with plane- and cylindrical-wavefronts are also introduced for calculating the associated travel times. Both synthetic and field data results show the improved quality of 3D PWLSM and cylindrical-wave least-squares migration (CWLSM) images compared to standard Kirchhoff migration with acceptable computational costs.

5.1 Introduction

In this Chapter, I apply both 3D plane-wave encoding and cylindrical-wave encoding to LSM Kirchhoff migration, and propose the methods of 3D plane-wave least-squares Kirchhoff migration (PWLSM) and cylindrical-wave least-squares Kirchhoff migration (CWLSM). To improve the robustness of PWLSM and CWLSM, 3D regularized plane-wave least-squares migration (RPWLSM) and cylindrical-wave least-squares Kirchhoff migration (CWLSM) are proposed in this chapter. The 3D finite-difference eikonal solvers for calculating plane-wave and cylindrical-wave travel times are also introduced.

For a narrow azimuth data set, the source sampling in the crossline direction is too sparse to support a 3D plane-wave encoding. As an alternative, an inline-only plane-wave encoding (also known as the cylindrical-wave encoding) can be adopted (Duquet and Lailly, 2006; Vigh and Starr, 2008). It also enjoys a computational saving of $\frac{N_{P_{inline}}}{N_{S_{inline}}}$ when $N_{S_{inline}}$ shots are encoded into $N_{P_{inline}}$ cylindrical-waves.

This chapter is organized into the following sections. The first section presents the theory of 3D PWLSM, 3D RPWLSM, CWLSM and RCWLSM. The next section presents synthetic and field data results of the proposed methods. A summary is provided in the last section.

5.2 Theory

5.2.1 3D Plane-wave Encoding

The plane-wave encoding technique can be defined as two steps: 1) apply a linear time shift to the shot gathers and 2) sum them up to form a plane-wave gather. The linear time shift is related to two ray parameters p_x and p_y in 3D geometry (Zhang et al., 2005; Duquet and Lailly, 2006; Vigh and Starr, 2008), while only one ray parameter p is needed for 2D.

The 3D plane-wave encoding is:

$$\begin{aligned} & d(x_s, y_s, z_s, t|x_p, y_p, z_s, 0) \\ & = \sum_{x_s} \sum_{y_s} d(x_r, y_r, z_r, t|x_s, y_s, z_s) * \delta(t - (p_x \cdot (x_s - x_0) + p_y \cdot (y_s - y_0))), \end{aligned} \quad (5.1)$$

where $d(x_r, y_r, z_r, t|x_s, y_s, z_s, 0)$ is the shot-domain data associated with the source position (x_s, y_s, z_s) and receiver position (x_r, y_r, z_r), p_x and p_y are the ray parameters in the crossline and inline directions, (x_0, y_0, z_0) is the reference source position, and $d(x_r, y_r, z_r, t|p_x, p_y, z_s)$ is the encoded plane-wave gather. As illustrated in Figure 5.1, the time shift $p_x \cdot (x_s - x_0) + p_y \cdot (y_s - y_0)$ is a linear function of the projected distances $(x_s - x_0)$ and $(y_s - y_0)$.

To simplify, $d(x_r, y_r, z_r, t|p_x, p_y, z_s)$ is written in vector notation as $d(\mathbf{r}, t|\mathbf{p})$, where $\mathbf{r}$ denotes the receiver position, $\mathbf{p}$ denotes the ray parameter vector, and the source depth z_s is omitted by assuming all the sources are on the surface ($z_s = 0$).

5.2.2 3D Plane-wave Kirchhoff Modeling and Migration

Kirchhoff modeling for 3D plane-wave can be expressed as:

$$d(\mathbf{r}, t|\mathbf{p}) = \int m(\mathbf{x}) \frac{\delta(t - \tau_{\mathbf{x},\mathbf{p}} - \tau_{\mathbf{rx}})}{A_{\mathbf{rx}}} d\mathbf{x}, \quad (5.2)$$

where $m(\mathbf{x})$ is the earth model, $\tau_{\mathbf{rx}}$ is the travel time from $\mathbf{x}$ to the receiver at $\mathbf{r}$, $\tau_{\mathbf{x},\mathbf{p}}$ is the travel time at $\mathbf{x}$ associated with the ray parameter vector $\mathbf{p}$, and $A_{\mathbf{rx}}$ is the geometric spreading term from $\mathbf{x}$ to $\mathbf{r}$[1].

[1]Here, the plane-wave source field does not suffer from geometric spreading.

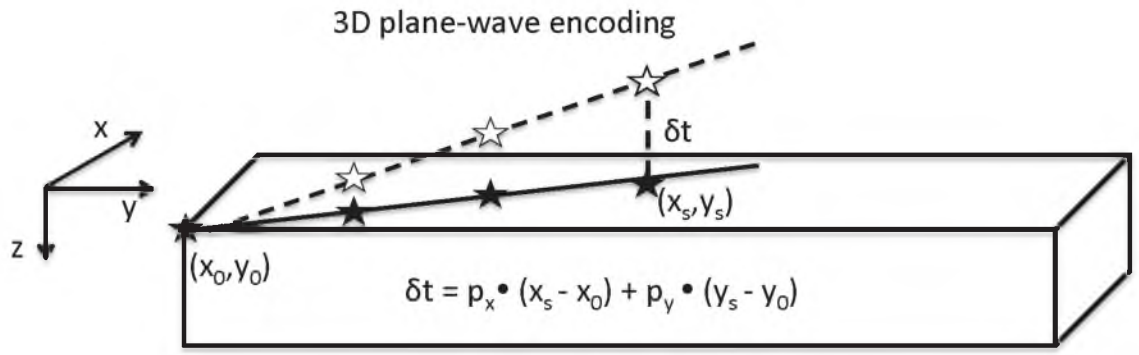

Figure 5.1. 3D plane-wave encoding.

Its adjoint operation (denoted as 3D plane-wave Kirchhoff migration (PWKM)) is

$$\widehat{m}(\mathbf{x}') = \int d\mathbf{r} \int d(\mathbf{r}, t|\mathbf{p}) \frac{\delta(t - \tau_{\mathbf{x}',\mathbf{p}} - \tau_{\mathbf{r}\mathbf{x}'})}{A_{\mathbf{r}\mathbf{x}'}} dt, \tag{5.3}$$

where $\widehat{m}(\mathbf{x}')$ is the plane-wave Kirchhoff migration image, and $\mathbf{x}'$ is the trial image point. The main difference between 2D plane-wave encoding (equations 4.3-4.4) and 3D plane-wave encoding (equations 5.2-5.3) is that the ray parameter p for 2D is now replaced by a ray parameter vector $\mathbf{p}$ that accounts for both the inline and crossline directions.

Equations 5.2 and 5.3 can be written in matrix-vector notation:

$$\widehat{\mathbf{d}}_{\mathbf{p}} = \widehat{\mathbf{L}}_{\mathbf{p}}\mathbf{m}, \tag{5.4}$$

$$\widehat{\mathbf{m}} = \widehat{\mathbf{L}}_{\mathbf{p}}^{\dagger}\widehat{\mathbf{d}}_{\mathbf{p}}. \tag{5.5}$$

Assuming there are N_{Px} and N_{Py} ray parameters in the x and y directions, we have

$$\widehat{\mathbf{d}} = \begin{bmatrix} \widehat{\mathbf{d}}_{1,1} \\ \widehat{\mathbf{d}}_{1,2} \\ \vdots \\ \widehat{\mathbf{d}}_{1,N_{Py}} \\ \widehat{\mathbf{d}}_{2,1} \\ \vdots \\ \widehat{\mathbf{d}}_{N_{Px},N_{Py}} \end{bmatrix} = \begin{bmatrix} \widehat{\mathbf{L}}_{1,1} \\ \widehat{\mathbf{L}}_{1,2} \\ \vdots \\ \widehat{\mathbf{L}}_{1,N_{Py}} \\ \widehat{\mathbf{L}}_{2,1} \\ \vdots \\ \widehat{\mathbf{L}}_{N_{Px},N_{Py}} \end{bmatrix} \mathbf{m} = \sum_{i=1}^{N_{Px}N_{Py}} \sum_{j=1} \widehat{\mathbf{L}}_{i,j}\mathbf{m} = \widehat{\mathbf{L}}\mathbf{m}, \tag{5.6}$$

$$\begin{aligned} \widehat{\mathbf{m}} &= \begin{bmatrix} \widehat{\mathbf{L}}_{1,1}^{\dagger} & \widehat{\mathbf{L}}_{1,2}^{\dagger} & \cdots & \widehat{\mathbf{L}}_{1,N_{Px}}^{\dagger} & \widehat{\mathbf{L}}_{2,1}^{\dagger} & \cdots & \widehat{\mathbf{L}}_{N_{Px},N_{Py}}^{\dagger} \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{d}}_{1,1} \\ \widehat{\mathbf{d}}_{1,2} \\ \vdots \\ \widehat{\mathbf{d}}_{1,N_{Py}} \\ \widehat{\mathbf{d}}_{2,1} \\ \vdots \\ \widehat{\mathbf{d}}_{N_{Px},N_{Py}} \end{bmatrix} \\ &= \sum_{i=1}^{N_{Px}N_{Py}} \sum_{j=1} \widehat{\mathbf{L}}_{i,j}^{\dagger}\widehat{\mathbf{d}}_{i,j} = \widehat{\mathbf{L}}^{\dagger}\widehat{\mathbf{d}}. \end{aligned} \tag{5.7}$$

5.2.3 3D Plane-wave Least-squares Kirchhoff Migration

A misfit function for the 3D plane-wave data can be expressed as:

$$f(\widehat{\mathbf{m}}) = \frac{1}{2}\|\widehat{\mathbf{L}}\widehat{\mathbf{m}} - \widehat{\mathbf{d}}\|_2^2, \tag{5.8}$$

and the 3D plane-wave least-squares Kirchhoff migration (PWLSM) can be defined as searching for a reflectivity model $\widehat{\mathbf{m}}$ to minimize this function. The numerical procedures of the preconditioned conjugate gradient method are similar to the 2D PWLSM introduced in section 4.2.3.

5.2.4 3D Regularized Plane-wave Least-squares Migration

Similar to section 4.2.4, 3D regularized plane-wave least-squares Kirchhoff migration (RPWLSM) is proposed to improve the robustness of PWLSM when the migration velocity model has bulk errors.

So, the ensemble of prestack images $\widehat{\mathbf{m}}$ and the associated modeling and migration operators can be defined as:

$$\widehat{\mathbf{m}} = \begin{bmatrix} \widehat{\mathbf{m}}_{1,1} \\ \widehat{\mathbf{m}}_{1,2} \\ \vdots \\ \widehat{\mathbf{m}}_{1,N_{Py}} \\ \widehat{\mathbf{m}}_{2,1} \\ \vdots \\ \widehat{\mathbf{m}}_{N_{Px},N_{Py}} \end{bmatrix}, \tag{5.9}$$

$$\widehat{\mathbf{d}} = \begin{bmatrix} \widehat{\mathbf{d}}_{1,1} \\ \widehat{\mathbf{d}}_{1,2} \\ \vdots \\ \widehat{\mathbf{d}}_{1,N_{Py}} \\ \widehat{\mathbf{d}}_{2,1} \\ \vdots \\ \widehat{\mathbf{d}}_{N_{Px},N_{Py}} \end{bmatrix} = \begin{bmatrix} \widehat{\mathbf{L}}_{1,1} & & & & & & \\ & \widehat{\mathbf{L}}_{1,2} & & & & & \\ & & \ddots & & & & \\ & & & \widehat{\mathbf{L}}_{1,N_{Py}} & & & \\ & & & & \widehat{\mathbf{L}}_{2,1} & & \\ & & & & & \ddots & \\ & & & & & & \widehat{\mathbf{L}}_{N_{Px},N_{Py}} \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{m}}_{1,1} \\ \widehat{\mathbf{m}}_{1,2} \\ \vdots \\ \widehat{\mathbf{m}}_{1,N_{Py}} \\ \widehat{\mathbf{m}}_{2,1} \\ \vdots \\ \widehat{\mathbf{m}}_{N_{Px},N_{Py}} \end{bmatrix} = \widehat{\mathcal{L}}\ \widehat{\mathbf{m}}, \tag{5.10}$$

and

$$\widehat{\mathbf{m}} = \begin{bmatrix} \widehat{\mathbf{m}}_{1,1} \\ \widehat{\mathbf{m}}_{1,2} \\ \vdots \\ \widehat{\mathbf{m}}_{1,N_{Py}} \\ \widehat{\mathbf{m}}_{2,1} \\ \vdots \\ \widehat{\mathbf{m}}_{N_{Px},N_{Py}} \end{bmatrix} = \begin{bmatrix} \widehat{\mathbf{L}}^{\dagger}_{1,1} & & & & & & \\ & \widehat{\mathbf{L}}^{\dagger}_{1,2} & & & & & \\ & & \ddots & & & & \\ & & & \widehat{\mathbf{L}}^{\dagger}_{1,N_{Py}} & & & \\ & & & & \widehat{\mathbf{L}}^{\dagger}_{2,1} & & \\ & & & & & \ddots & \\ & & & & & & \widehat{\mathbf{L}}^{\dagger}_{N_{Px},N_{Py}} \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{d}}_{1,1} \\ \widehat{\mathbf{d}}_{1,2} \\ \vdots \\ \widehat{\mathbf{d}}_{1,N_{Py}} \\ \widehat{\mathbf{d}}_{2,1} \\ \vdots \\ \widehat{\mathbf{d}}_{N_{Px},N_{Py}} \end{bmatrix} = \widehat{\mathcal{L}}^{\dagger}\ \widehat{\mathbf{d}}. \tag{5.11}$$

A new misfit function is defined as

$$f(\widehat{\mathbf{m}}) \ = \ \frac{1}{2}\|\widehat{\mathcal{L}}\widehat{\mathbf{m}} - \widehat{\mathbf{d}}\|_2^2 + \mathbf{R}. \tag{5.12}$$

The regularization term $\mathbf{R}$ is defined as a function that penalizes the difference between migration images computed with slightly different incidence angles:

$$\mathbf{R} = \frac{1}{2}\gamma \sum_{i=2}^{N_{Px}-1} \sum_{j=2}^{N_{Py}-1} ||\widehat{\mathbf{m}}_{i+1,j} + \widehat{\mathbf{m}}_{i-1,j} + \widehat{\mathbf{m}}_{i,j+1} + \widehat{\mathbf{m}}_{i,j-1} - 4\widehat{\mathbf{m}}_{i,j}||_2^2, \tag{5.13}$$

where γ is the damping coefficient determined by trial-and-error testing. We denote this method as 3D regularized plane-wave least-squares Kirchhoff migration (RPWLSM). The numerical procedures for RPWLSM are similar to the one for 2D RPWLSM (see section 4.2.4).

5.2.5 Cylindrical-wave Encoding

For narrow azimuth streamer data, the source sampling in the crossline direction is too sparse to enable plane-wave encoding with p_x. An inline-only plane-wave encoding is adopted, which is also known as the cylindrical-wave encoding[2]. If x and y denote the crossline and inline directions, respectively, a cylindrical-wave encoding can be expressed as:

$$d(x_r, y_r, z_r, t; x_s, p_y, z_s) = \sum_{y_s} d(x_r, y_r, z_r, t; x_s, y_s, z_s) * \delta(t - p_y \cdot (y_s - y_0)), \tag{5.14}$$

where all the shots are at the same crossline location x_s but different inline locations $\sum_{i=1}^{N_{Sy}} y_{s_i}$, and y_0 denotes the inline coordinate of the reference source. The time-shift function $\delta(p_y \cdot (y_s - y_0))$ is only related to the ray parameter p_y and the inline distance of $(y_s - y_0)$. Figure 5.2 illustrates the cylindrical-wave encoding. Compared to the 2D plane-wave encoding in Figure 4.1, it is found that cylindrical-wave encoding can be defined as applying the 2D plane-wave encoding to line sources in the 3D geometry. To simplify, we also write the cylindrical-wave data as $d(\mathbf{r}, t; p_y)$, where $\mathbf{r}$ denotes the receiver position, and the source x and z coordinates are omitted by assuming all the sources are on the surface and with the same crossline position ($z_s = 0, x_s = const.$).

5.2.6 Cylindrical-wave Kirchhoff Modeling, Migration and Least-squares Migration

The 3D cylindrical-wave Kirchhoff modeling can be expressed as:

$$d(\mathbf{r}, t|p_y) = \int m(\mathbf{x}) \frac{\delta(t - \tau_{\mathbf{x},p_y} - \tau_{\mathbf{rx}})}{A_{\mathbf{rx}} A_{\mathbf{x},p_y}} d\mathbf{x}, \tag{5.15}$$

where $m(\mathbf{x})$ is the earth model, $\tau_{\mathbf{rx}}$ is the travel time from $\mathbf{x}$ to the receiver at $\mathbf{r}$, $\tau_{\mathbf{x},p_y}$ is the travel time at $\mathbf{x}$ associated with the ray parameter p_y, $A_{\mathbf{rx}}$ is the geometric spreading term from $\mathbf{x}$ to $\mathbf{r}$, and $A_{\mathbf{x},p_y}$ is the cylindrical-wave geometric spreading term[3].

---

[2]Exploding a series of sources placed in a line at the same time will produce a cylindrical wavefield in the 3D space. If the sources are exploded with a linear time delay, a conic wavefield will be produced instead.

[3]It is a 2D geometric spreading term, which can be estimated by $A_{\mathbf{x},p_y} = \frac{1}{\sqrt{(x-x_s)^2+(z-z_s)^2}}$, where x_s and z_s are the coordinates of the cylindrical-wave sources.

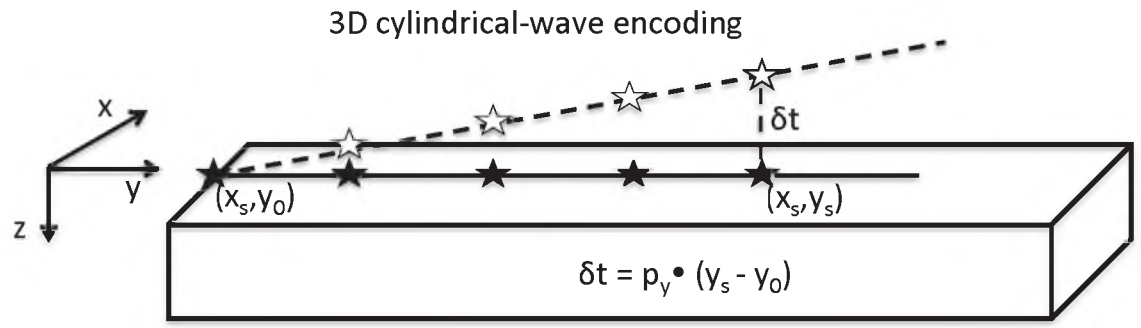

Figure 5.2. 3D cylindrical-wave encoding for a nonzero p_y value.

Its adjoint operation (denoted as 3D cylindrical-wave Kirchhoff migration) can be expressed as

$$\breve{m}(\mathbf{x}') = \int d\mathbf{r} \int d(\mathbf{r}, t|p_y) \frac{\delta(t - \tau_{\mathbf{x}',p_y} - \tau_{\mathbf{r}\mathbf{x}'})}{A_{\mathbf{r}\mathbf{x}'} A_{\mathbf{x},p_y}} dt, \tag{5.16}$$

where $\breve{m}(\mathbf{x}')$ is the cylindrical-wave Kirchhoff migration image, and $\mathbf{x}'$ is the trial image point.

Equations 5.15 and 5.16 can be written in matrix-vector notation:

$$\breve{\mathbf{d}}_{p_y} = \breve{\mathbf{L}}_{p_y} \mathbf{m}, \tag{5.17}$$

$$\breve{\mathbf{m}} = \breve{\mathbf{L}}^{\dagger}_{p_y} \breve{\mathbf{d}}_{p_y}, \tag{5.18}$$

and for N_{Py} cylindrical-waves, we have

$$\breve{\mathbf{d}} = \begin{bmatrix} \breve{\mathbf{d}}_1 \\ \breve{\mathbf{d}}_2 \\ \vdots \\ \breve{\mathbf{d}}_{N_{Py}} \end{bmatrix} = \begin{bmatrix} \breve{\mathbf{L}}_1 \\ \breve{\mathbf{L}}_2 \\ \vdots \\ \breve{\mathbf{L}}_{N_{Py}} \end{bmatrix} \mathbf{m} = \sum_{i=1}^{N_{Py}} \breve{\mathbf{L}}_i \mathbf{m} = \breve{\mathbf{L}} \mathbf{m}, \tag{5.19}$$

$$\breve{\mathbf{m}} = \begin{bmatrix} \breve{\mathbf{L}}^{\dagger}_1 & \breve{\mathbf{L}}^{\dagger}_2 & \cdots & \breve{\mathbf{L}}^{\dagger}_{N_{Py}} \end{bmatrix} \begin{bmatrix} \breve{\mathbf{d}}_1 \\ \breve{\mathbf{d}}_2 \\ \vdots \\ \breve{\mathbf{d}}_{N_{Py}} \end{bmatrix} = \sum_{i=1}^{N_{Py}} \breve{\mathbf{L}}^{\dagger}_i \breve{\mathbf{d}}_i = \breve{\mathbf{L}}^{\dagger} \breve{\mathbf{d}}. \tag{5.20}$$

The cylindrical-wave least-squares Kirchhoff migration (CWLSM) can be defined as searching for a reflectivity model $\breve{\mathbf{m}}$ to minimize the cylindrical-wave domain misfit function

$$f(\breve{\mathbf{m}}) = \frac{1}{2} \| \breve{\mathbf{L}} \breve{\mathbf{m}} - \breve{\mathbf{d}} \|_2^2. \tag{5.21}$$

The numerical implementation of the preconditioned conjugate gradient method is similar to that of 2D PWLSM introduced in section 4.2.3.

To improve the robustness of CWLSM, we assume each cylindrical-wave gather $\breve{\mathbf{d}}_i$ is associated with its own migration image $\breve{\mathbf{m}}_i$, so an ensemble of the prestack images $\underline{\breve{\mathbf{m}}}$ and the associated modeling and migration operators are defined as:

$$\underline{\breve{\mathbf{m}}} = \begin{bmatrix} \breve{\mathbf{m}}_1 \\ \breve{\mathbf{m}}_2 \\ \vdots \\ \breve{\mathbf{m}}_{N_{Py}} \end{bmatrix}, \tag{5.22}$$

and the modeling and migration equations can be expressed as

$$\breve{\mathbf{d}} = \begin{bmatrix} \breve{\mathbf{d}}_1 \\ \breve{\mathbf{d}}_2 \\ \vdots \\ \breve{\mathbf{d}}_{N_{Py}} \end{bmatrix} = \begin{bmatrix} \breve{\mathbf{L}}_1 & & & \\ & \breve{\mathbf{L}}_2 & & \\ & & \vdots & \\ & & & \breve{\mathbf{L}}_{N_{Py}} \end{bmatrix} \begin{bmatrix} \breve{\mathbf{m}}_1 \\ \breve{\mathbf{m}}_2 \\ \vdots \\ \breve{\mathbf{m}}_{N_{Py}} \end{bmatrix} = \breve{\mathfrak{L}} \, \underline{\breve{\mathbf{m}}}, \tag{5.23}$$

and

$$\check{\underline{\mathbf{m}}} = \begin{bmatrix} \check{\mathbf{m}}_1 \\ \check{\mathbf{m}}_2 \\ \vdots \\ \check{\mathbf{m}}_{N_{Py}} \end{bmatrix} = \begin{bmatrix} \check{\mathbf{L}}_1^\dagger & & & \\ & \check{\mathbf{L}}_2^\dagger & & \\ & & \vdots & \\ & & & \check{\mathbf{L}}_{N_{Py}}^\dagger \end{bmatrix} \begin{bmatrix} \check{\mathbf{d}}_1 \\ \check{\mathbf{d}}_2 \\ \vdots \\ \check{\mathbf{d}}_{N_{Py}} \end{bmatrix} = \check{\mathfrak{L}}^\dagger \, \check{\mathbf{d}}. \tag{5.24}$$

A new misfit function is defined as

$$f(\check{\underline{\mathbf{m}}}) = \frac{1}{2}\|\check{\mathfrak{L}}\check{\underline{\mathbf{m}}} - \check{\mathbf{d}}\|_2^2 + \mathbf{R}, \tag{5.25}$$

and the regularization term is defined as a function that penalizes the difference between migration images computed with slightly different inline incidence angles

$$\mathbf{R} = \frac{1}{2}\gamma \sum_{i=1}^{N_{Py}-1} ||\check{\mathbf{m}}_{i+1} - \check{\mathbf{m}}_i||_2^2. \tag{5.26}$$

We denote this method as regularized cylindrical-wave least-squares migration (RCWLSM). The preconditioned conjugate gradient implementation is similar to the 2D RPWLSM one (see section 4.2.4).

5.2.7 3D Plane- and Cylindrical-waves Eikonal Solvers

In this section, I introduce the FD solvers to the eikonal equation to calculate the 3D plane-wave and cylindrical-wave travel times on the 3D expanding wavefront FD scheme. The 3D eikonal equation can be expressed as

$$(\frac{\partial T}{\partial x})^2 + (\frac{\partial T}{\partial y} + (\frac{\partial T}{\partial z})^2 = s(x,y,z)^2, \tag{5.27}$$

where $T(x,y,z)$ is the arrival time for seismic energy through a medium with the slowness $s(x,y,z)$.

The 2D numerical procedures for the expanding wavefront method for a point source (see section 4.2.5) can be easily extended to a 3D geometry, and the main differences are:

1. the travel times at grids around the source point (x_s, y_s, z_s) are now within a user-defined $N \times N \times N$ cubic of gridpoints,

2. the 3D FD stencils are used instead of 2D ones (see Appendix B).

3. The boundary values of the eikonal equation for a point source can be defined as

$$T(x_s, y_s, z_s) = 0. \tag{5.28}$$

To calculate the plane-wave and cylindrical-wave travel times, the boundary values are defined as

$$\sum_{i=1}^{Nx}\sum_{j=1}^{Ny} T(x_i, y_j, 0) = p_x \cdot (x_i - x_0) + p_y \cdot (y_j - y_0), \tag{5.29}$$

and

$$\sum_{j=1}^{Ny} T(x_s, y_j, 0) = py \cdot (y_j - y_0), \tag{5.30}$$

where x_s is the crossline coordinate of the cylindrical sources.

The FD eikonal solver for 3D plane-waves is tested on the SEG Overthrust model (in Figure 5.3). Figures 5.4-5.5 show the plane-wave travel time contours associated with two ray parameter vectors $\mathbf{p}_1 = (-0.02, 0.02)$ and $\mathbf{p}_1 = (0, 0.0048)$ ms/m. The FD eikonal solver for 3D cylindrical-waves is also tested on the SEG Overthrust model (in Figure 5.6), but with a different model size. Figures 5.7-5.8 show the cylindrical-wave travel time contours associated with the inline ray parameter values of -0.015 ms/m and 0.026 ms/m.

5.3 Numerical Tests

5.3.1 Synthetic Test of 3D Plane-wave Data

The 3D PWLSM and RPWLSM methods are tested on the SEG Overthust model (as shown in Figure 5.3), where the model size is $5 \times 5 \times 2.5$ km with a gridpoint interval of 25 m. The acquisition system is designed to simulate an ocean bottom stations (OBS) acquisition geometry. Four hundred receivers are evenly planted on the water bottom at the depth of 375 m with a receiver interval of 250 m in both the inline and cross line directions. Fifty sail lines with a 100 m crossline interval are designed to shoot on the surface, and each sail line consists of 99 shots with a 50 m inline interval, so the total number of shots is 4950. Synthetic data are computed by a finite-difference solver to the 3D acoustic wave-equation with a 15 Hz peak-frequency Ricker wavelet. Figure 5.9 shows the CSG-domain KM images.

The total of 4950 CSGs are encoded with 31 p_x and 31 p_y ray parameters to produce 961 plane-wave gathers. The range of ray parameters for both p_x and p_y is between -0.11 and 0.11 ms/m, which indicate the maximum surface shooting angle is 30°. Figures 5.10-5.12 show, respectively, the images of PWKM, PWLSM and RPWLSM (both PWLSM and RPWLSM are stopped after 10 iterations). Compared to PWKM, both PWLSM and RPWLSM can provide high quality images (indicated by the green circles in Figures 5.10-5.12). Figure 5.13 shows 10 common image gathers (CIG) at the crossline position $x = 2.5$ km, but different inline positions. Each CIG is with the same ray parameter $p_y = 0$, but p_x ranges from -0.11 and 0.11 ms/m. Figure 5.14 shows 10 common image gathers (CIG) at

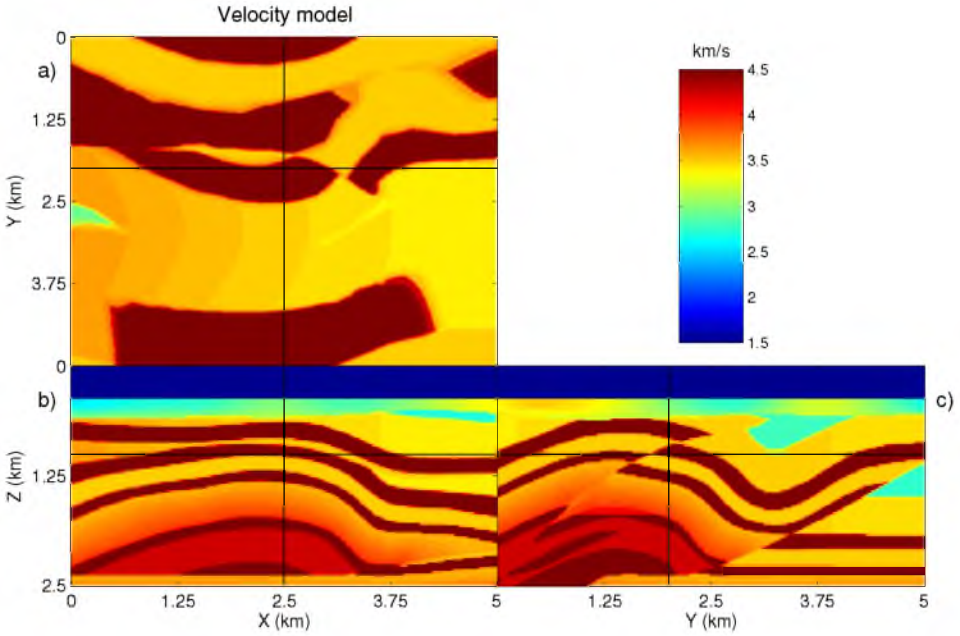

Figure 5.3. Overthrust velocity model for testing the 3D plane-wave eikonal solver, where the model size is $5 \times 5 \times 2.5$ km with a 25 m gridpoint interval: a) XY plane at $z = 1$ km, b) XZ plane at $y = 2.5$ km and c) YZ plane at $x = 2.5$ km.

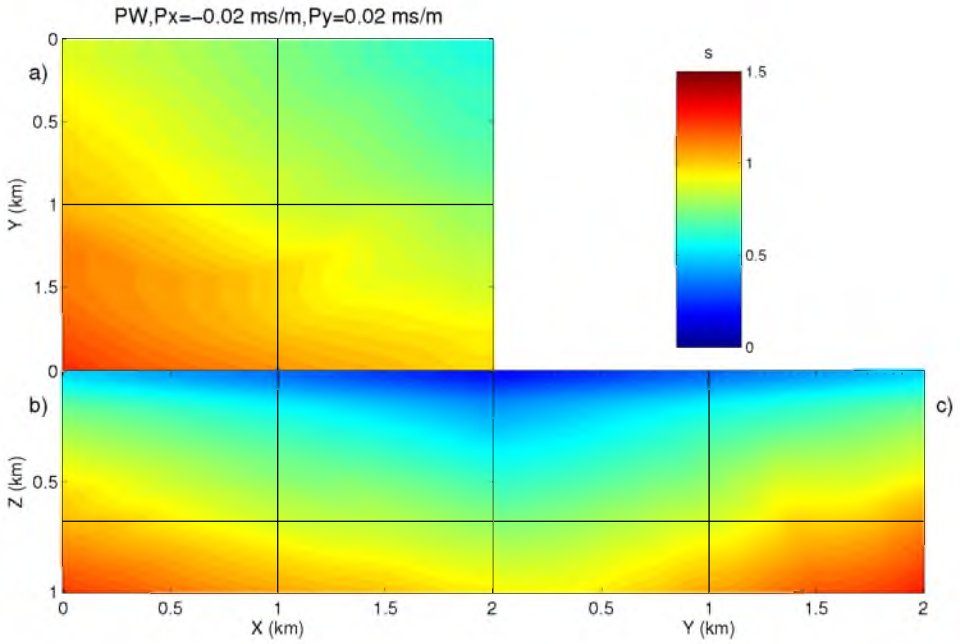

Figure 5.4. 3D plane-wave travel times with ray parameter $p_x = -0.02$ and $p_y = 0.02$ ms/m: a) XY plane at $z = 1$ km, b) XZ plane at $y = 2.5$ km and c) YZ plane at $x = 2.5$ km.

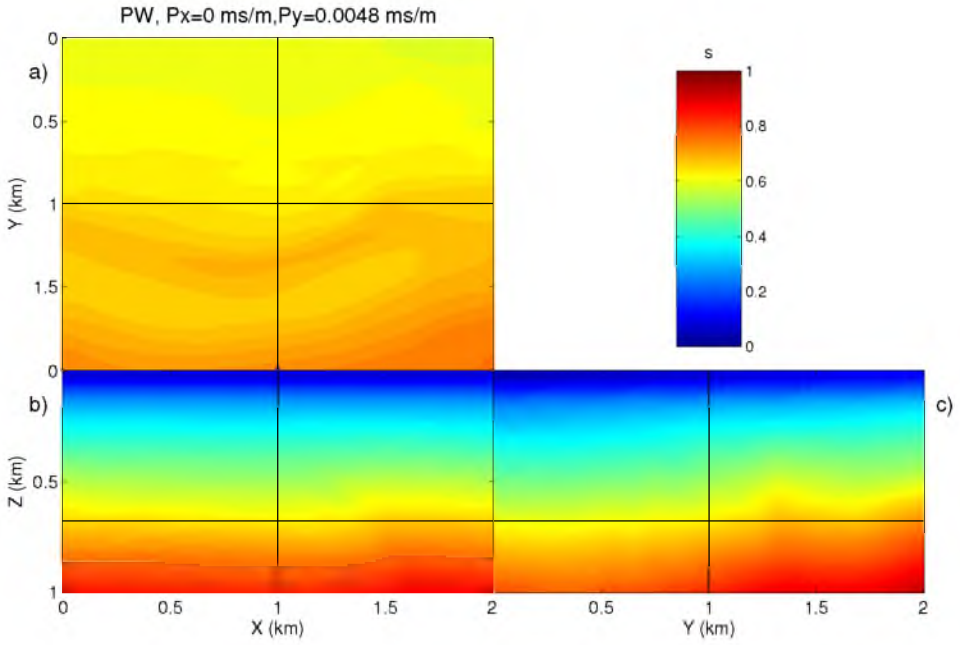

Figure 5.5. 3D plane-wave travel times with ray parameters $p_x = 0$ and $p_y = 0.0048$ ms/m: a) XY plane at $z = 1$ km, b) XZ plane at $y = 2.5$ km and c) YZ plane at $x = 2.5$ km.

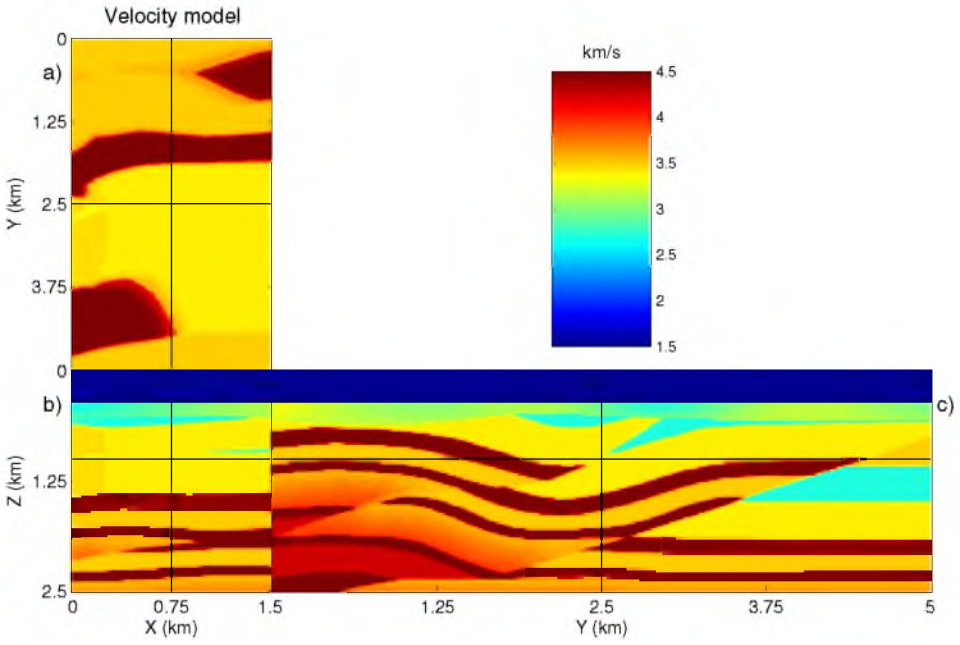

Figure 5.6. Overthrust velocity for testing the 3D cylindrical-wave eikonal solver, where the model size is $1.5 \times 5 \times 2.5$ km with a 25 m gridpoint interval: a) XY plane at $z = 1$ km, b) XZ plane at $y = 2.5$ km and c) YZ plane at $x = 0.75$ km.

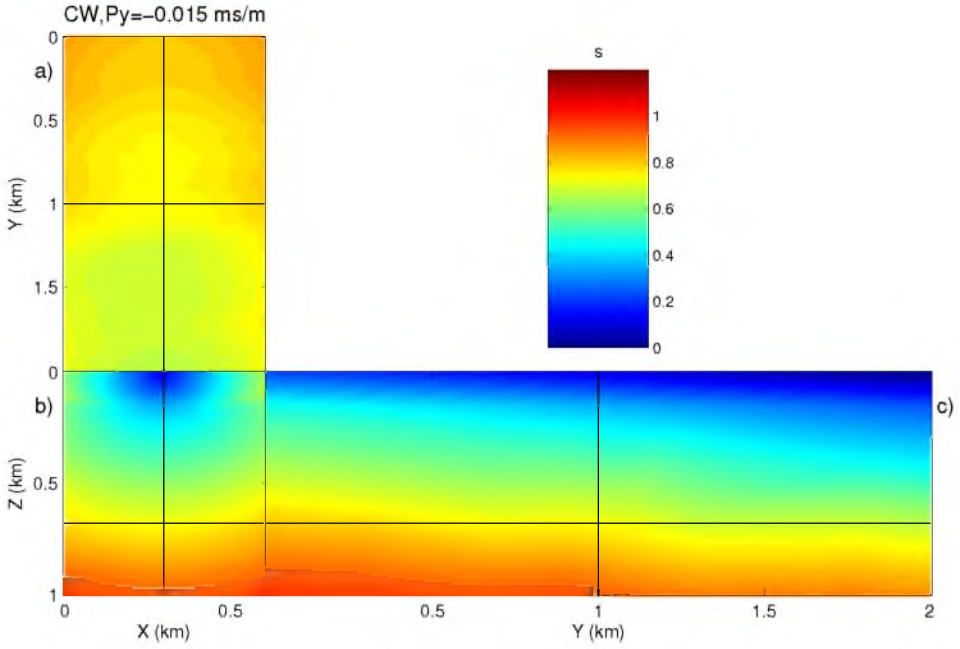

Figure 5.7. 3D cylindrical-wave travel times with ray parameters $p_y = -0.015$ ms/m: a) XY plane at $z = 1$ km, b) XZ plane at $y = 2.5$ km and c) YZ plane at $x = 0.75$ km.

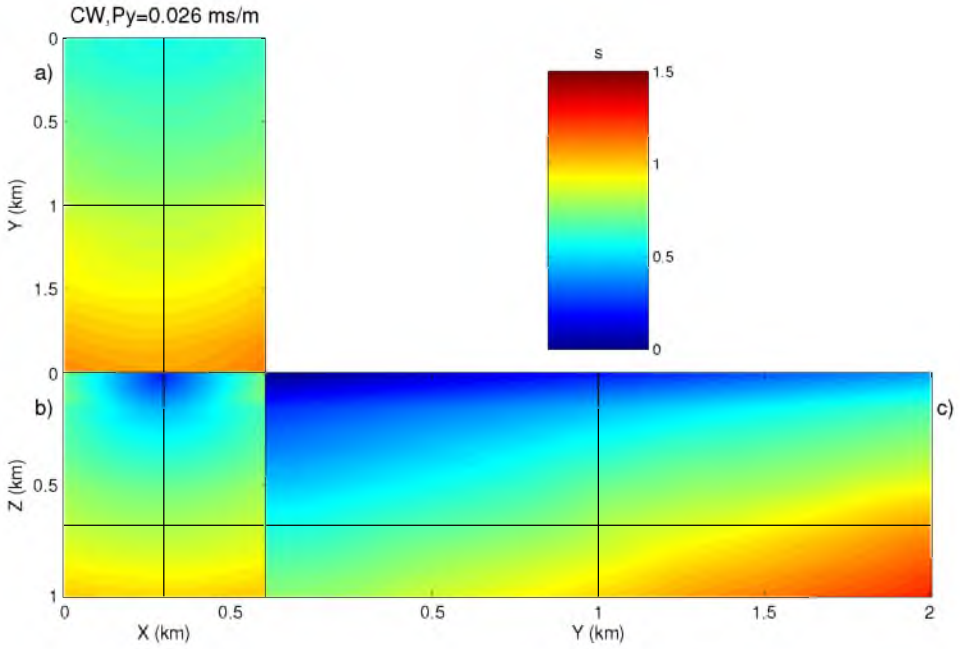

Figure 5.8. 3D cylindrical-wave travel times with ray parameters $p_y = 0.026$ ms/m: a) XY plane at $z = 1$ km, b) XZ plane at $y = 2.5$ km and c) YZ plane at $x = 0.75$ km.

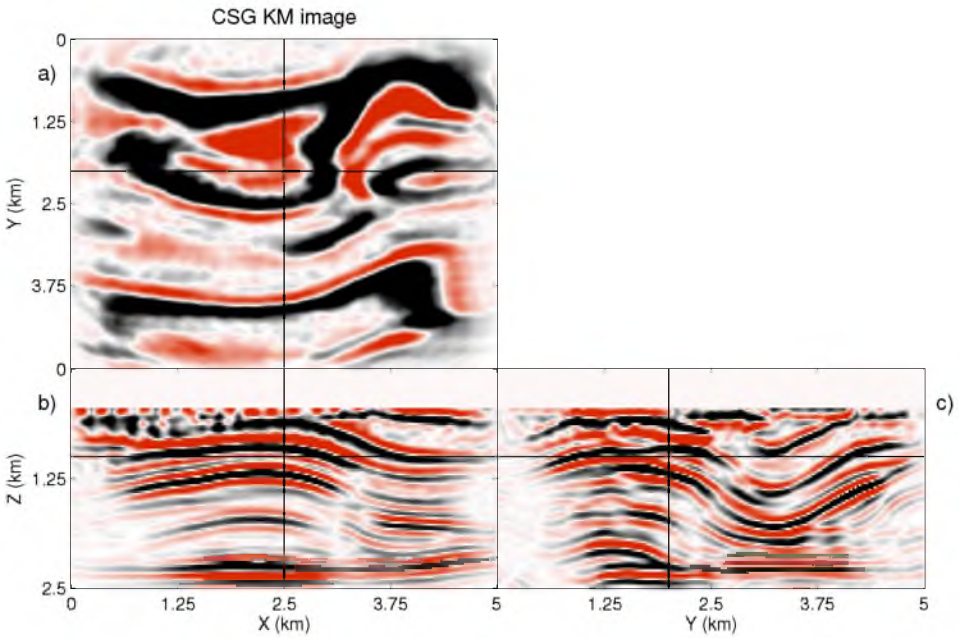

Figure 5.9. CSG-domain KM images: a) XY plane at $z = 1$ km, b) XZ plane at $y = 1.875$ km and c) YZ plane at $x = 2.5$ km.

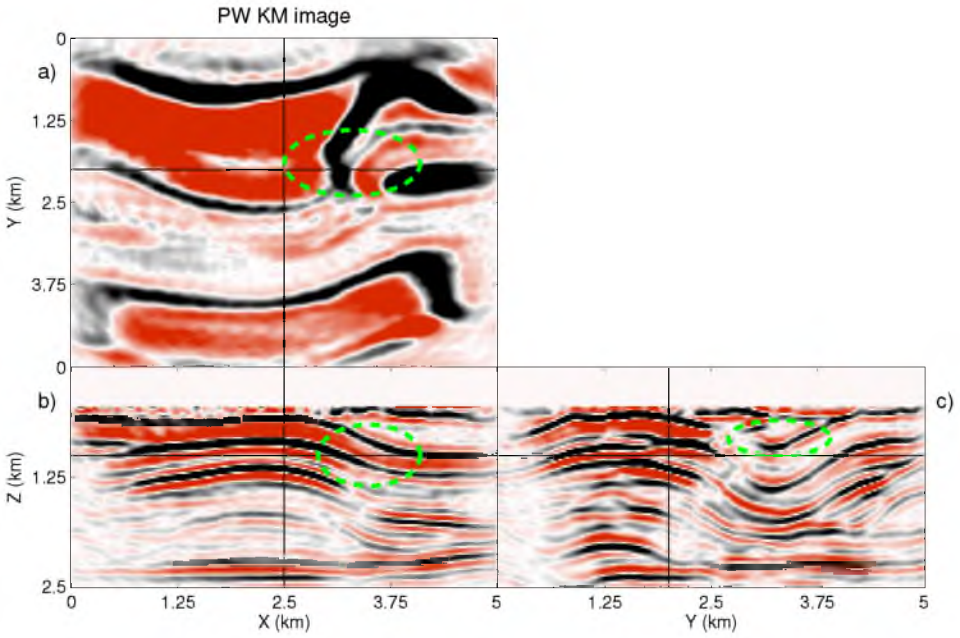

Figure 5.10. PW-domain KM images: a) XY plane at $z = 1$ km, b) XZ plane at $y = 1.875$ km and c) YZ plane at $x = 2.5$ km.

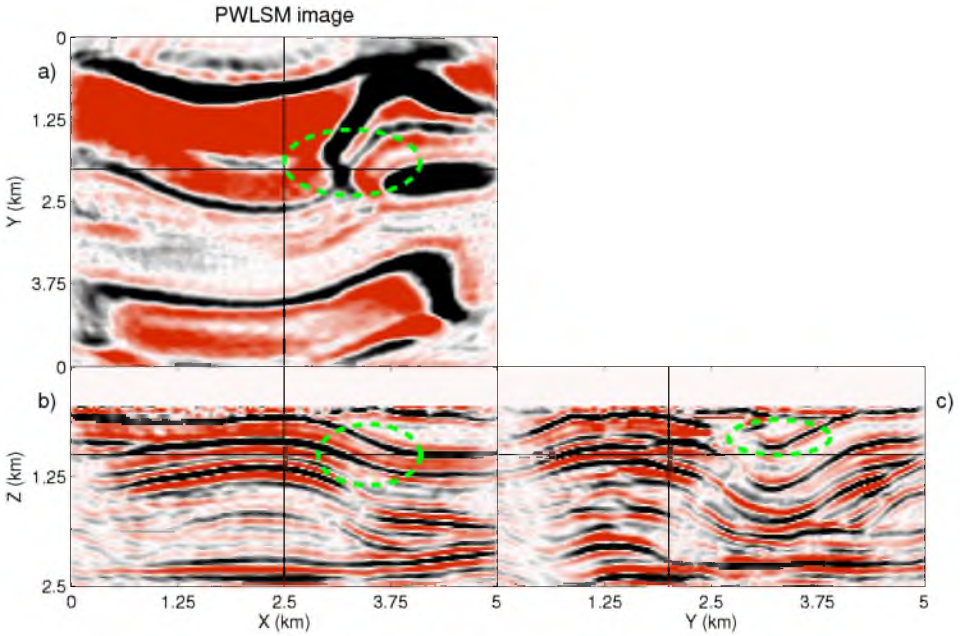

Figure 5.11. PWLSM images: a) XY plane at $z = 1$ km, b) XZ plane at $y = 1.875$ km and c) YZ plane at $x = 2.5$ km.

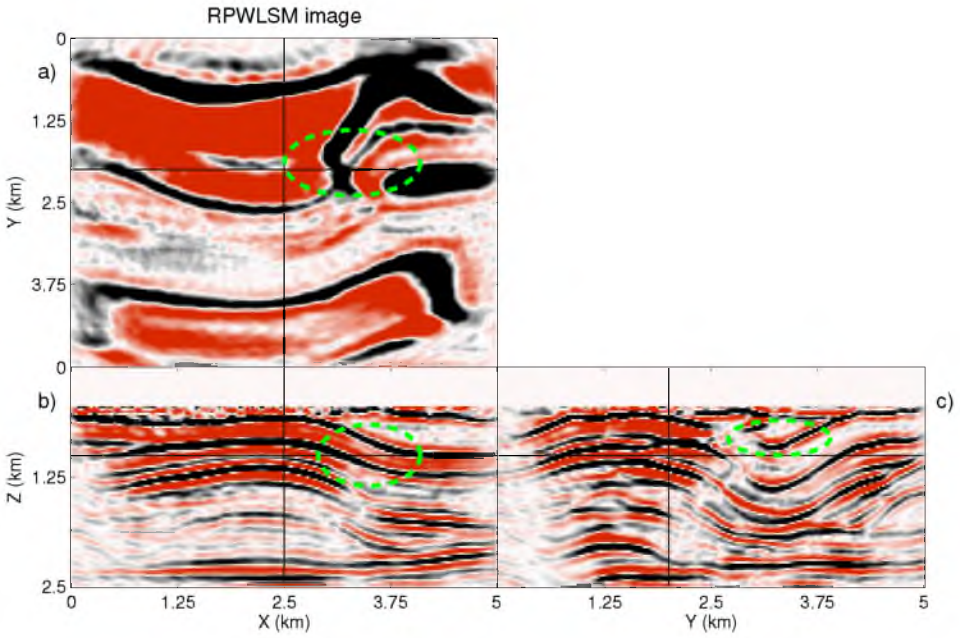

Figure 5.12. RPWLSM images: a) XY plane at $z = 1$ km, b) XZ plane at $y = 1.875$ km and c) YZ plane at $x = 2.5$ km.

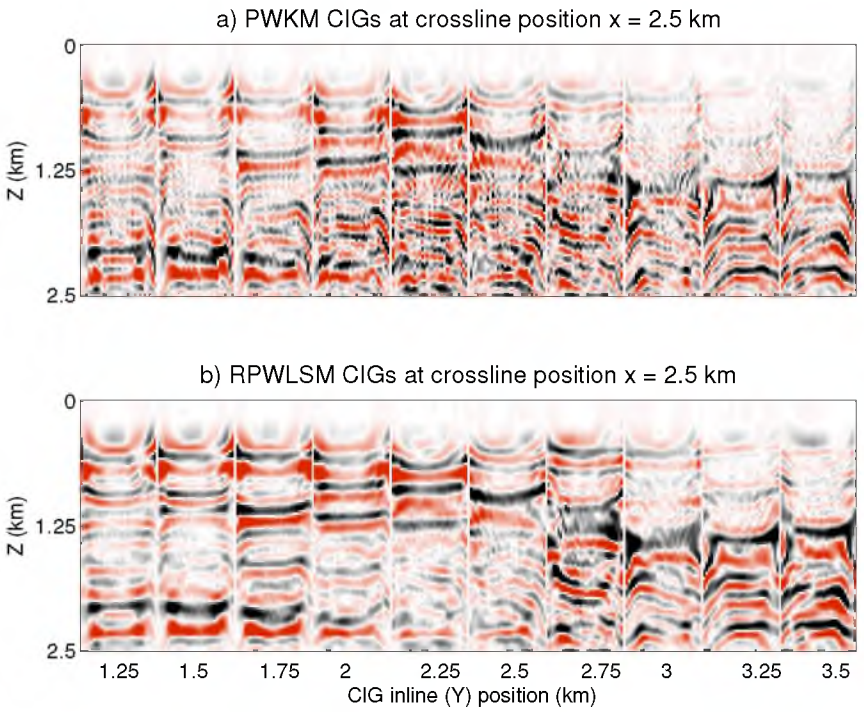

Figure 5.13. CIGs of different inline (y) positions at $x = 2.5$ km for: a) PWKM image and b) RPWLSM image.

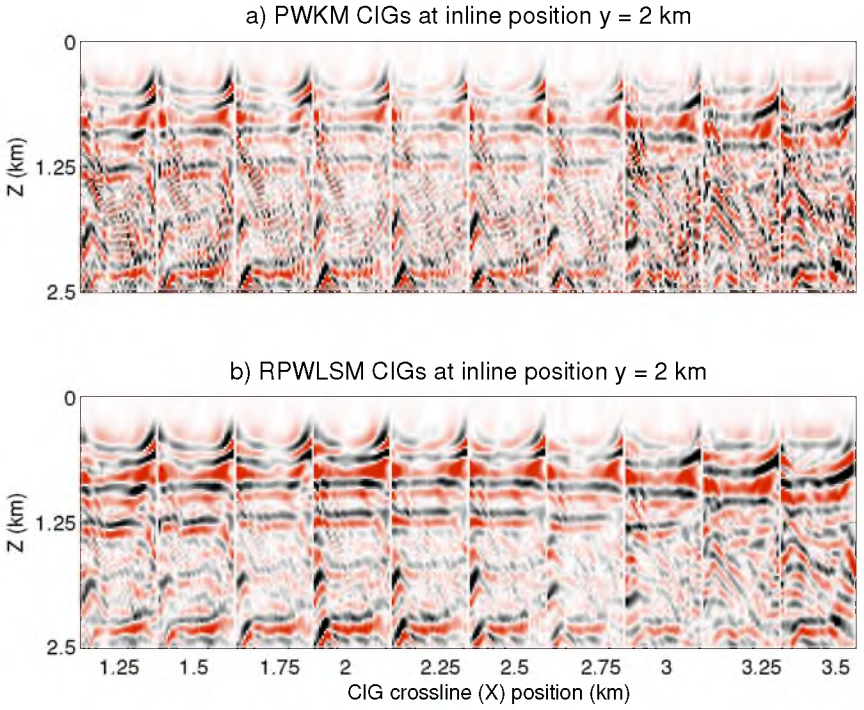

Figure 5.14. CIGs of different crossline (x) positions at $y = 2$ km for: a) PWKM image and b) RPWLSM image.

the inline position $y = 2$ km, but different crossline positions. Each CIG is with the same ray parameter $p_x = 0$, but p_y ranges from -0.11 and 0.11 ms/m. It is found that RPWLSM can significantly decrease the migration artifacts in prestack images, compared to PWKM.

Figure 5.15 compares the convergence rates of PWLSM and RPWLSM, and shows that RPWLSM converges faster than PWLSM. Although the migration velocity model is accurate, the original data are generated with the FD solution to the wave equation ($\mathbf{d}_i = \mathbf{A}_i(v)$, where $\mathbf{A}_i$ is the nonlinear FD modeling operator), but the modeling and migration operators in the LSM algorithms are Kirchhoff operators $\mathbf{L}_i$ and $\mathbf{L}_i^\dagger$; applying $\mathbf{L}_i^\dagger$ to $\mathbf{d}_i$ will introduce an inconsistency between the modeled and the observed data. This also indicates that RPWLSM may have a better convergence rate than PWLSM when they are applied to field data.

Table 5.1 shows that PWLSM and RPWLSM can reduce the CPU and IO costs of CSG LSM to an acceptable level, which is about 4 times that for conventional CSG-domain LSM[4].

5.3.2 Synthetic Test of 3D Cylindrical-wave Data

The 3D CWLSM and RCWLSM methods are tested on the 3D SEG Overthust model (as shown in Figure 5.6), where the model size is 1.5×2.5 km with a gridpoint inveval of 25 m. There are 201 shots with an inline shot interval of 25 m, and each shot is recorded by 8 receiver lines separated by a 100 m crossline interval, where each receiver line consists of 201 receivers with a 25 m inline interval. Synthetic data are computed by a finite-difference solver to the 3D acoustic wave-equation with a 15 Hz peak-frequency Ricker wavelet. Figure 5.16 shows the CSG-domain KM images. Since there are only 1 sail and 8 receiver lines in the crossline section, the resolution is poor in both the XZ (Figure 5.16b) and XY (Figure 5.16a) planes, and only the inline YZ plane (Figure 5.16c) will be used for comparison.

A total of 201 CSGs are encoded into 51 p_y ray parameters with a range of -0.11 and 0.11 ms/m. Figure 5.17a-c shows, respectively, the images of CWKM, CWLSM and RCWLSM (both CWLSM and RCWLSM are stopped after 10 iterations). Compared to CWKM, both CWLSM and RCWLSM can provide high quality images. Figure 5.18a-b show that the CIGs produced by RCWLSM have higher resolution than those produced by CWKM, and these high resolution CIGs enhance the opportunity for migration velocity analysis. Figure 5.19 shows that RCWLSM converges faster than CWLSM, and this is

[4]The CPU and IO costs for CSG LSM are accurately estimated from the algorithm.

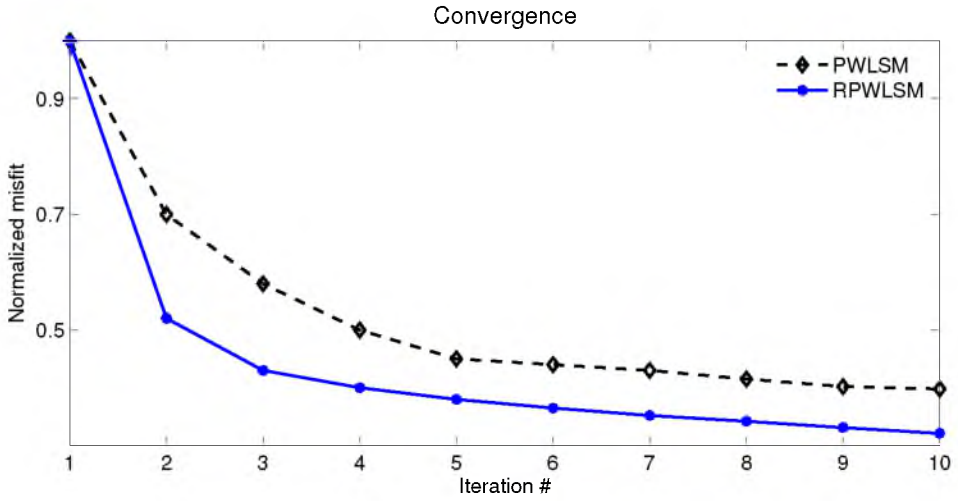

Figure 5.15. Convergent curves of PWLSM and RPWLSM. Black line is for PWLSM and blue is for RPWLSM.

Table 5.1. Comparison of image quality and CPU, IO costs for KM, PWKM and PWLSM for the Overthrust synthetic data test.

	CPU expense	**IO demand**
CSG KM	1	1
PWKM	0.194	0.194
CSG LSM	21	21
PWLSM	4.07	4.07
RPWLSM	4.07	4.07

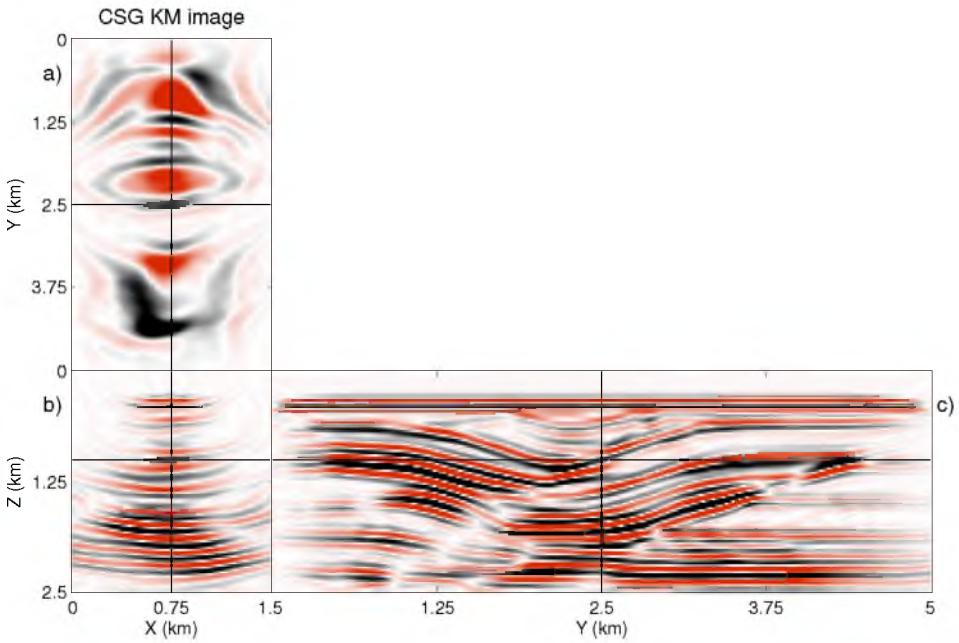

Figure 5.16. CSG-domain KM images: a) XY plane at $z = 1$ km, b) XZ plane at $y = 2.5$ km and c) YZ plane at $x = 2.5$ km.

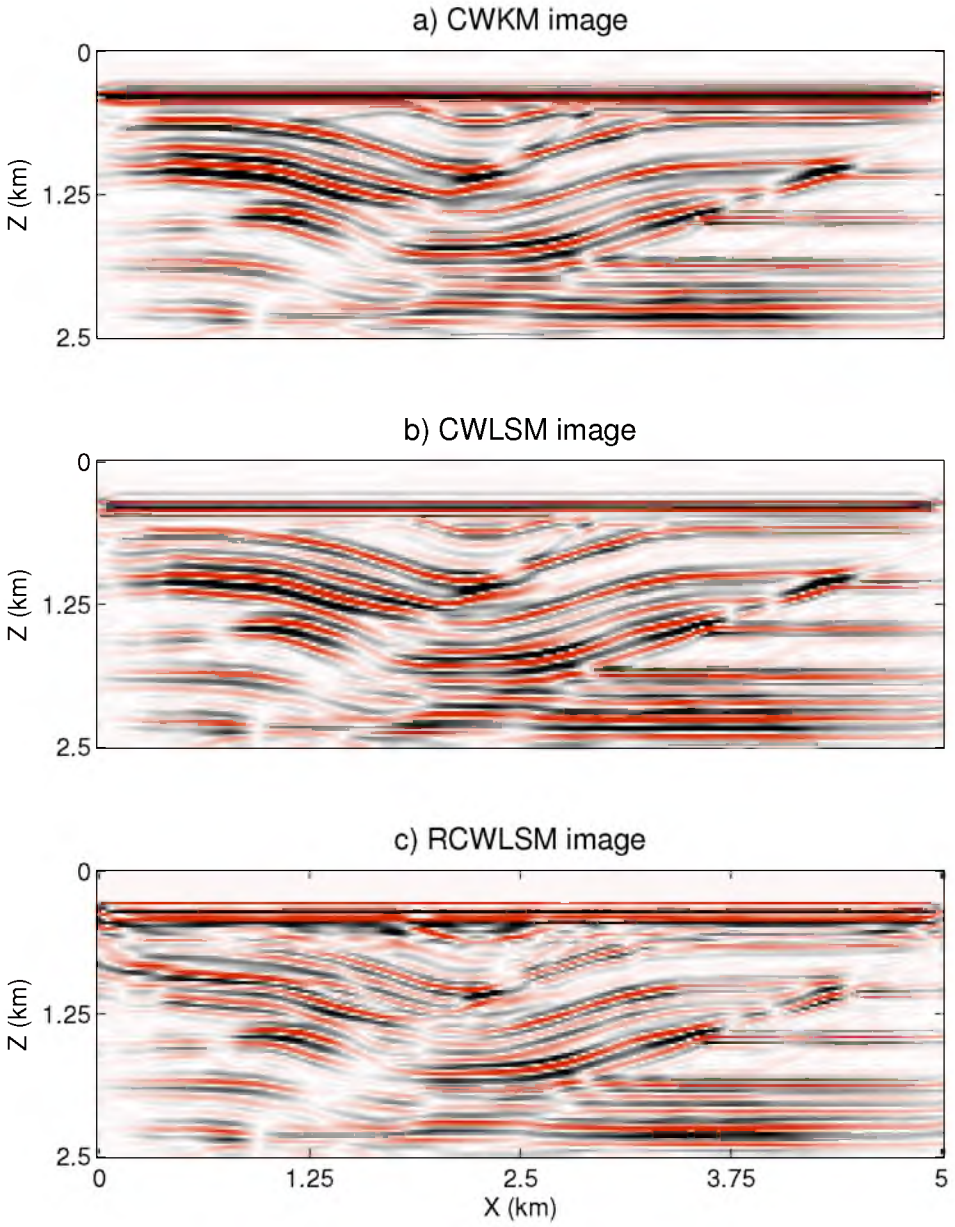

Figure 5.17. Images produced by: a) CWKM, b) CWLSM and c) RCWLSM. Both CWLSM and RCWLSM are stopped after 10 iterations.

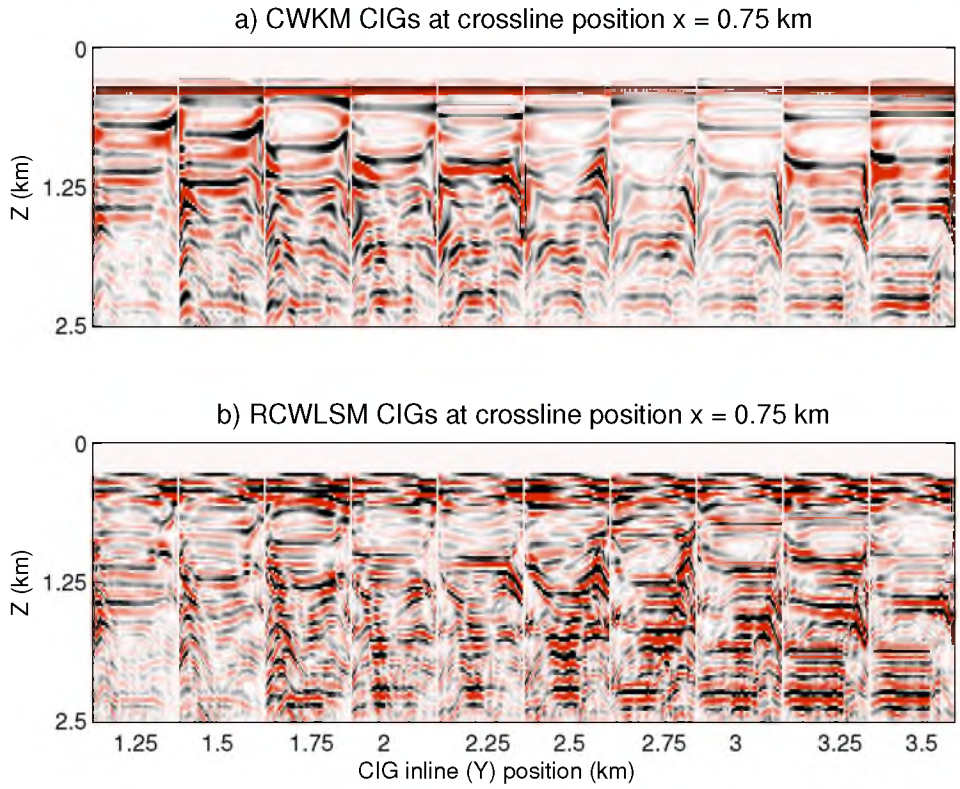

Figure 5.18. CIGs produced by: a) CWKM and b) RCWLSM.

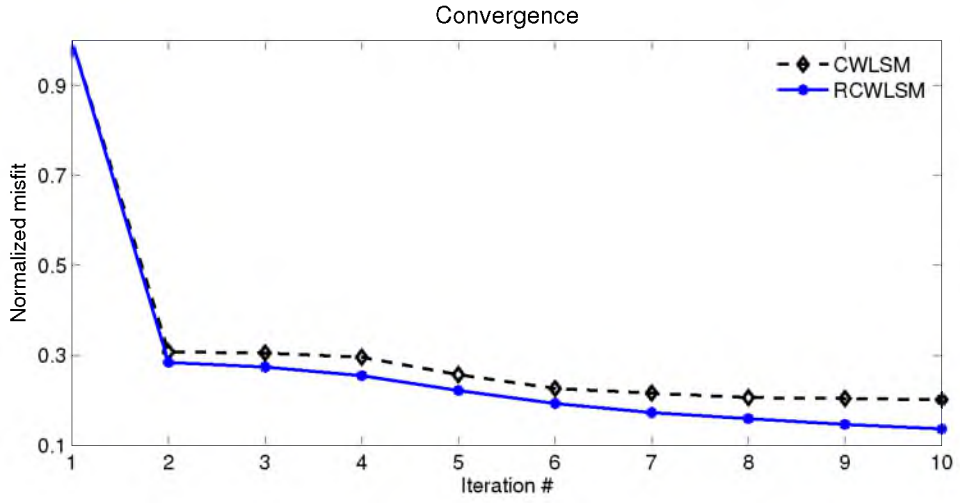

Figure 5.19. Convergent curves of CWLSM and RCWLSM. Black line is for CWLSM and blue is for RCWLSM.

consistent with the 3D PWLSM test.

5.3.3 Field Test of 3D Cylindrical-wave Data

The proposed methods are also tested on 3D GOM data, where the model size is $0.8 \times 8.3 \times 5$ km with a gridpoint inveval of 12.5 m. The migration velocity is shown in Figure 5.20. There are 251 shots with a shot interval of 25 m, and each shot is recorded by 6 streamer lines separated by a 75 m crossline interval, where each receiver line consists of 324 receivers with a 25 m inline interval. Figure 5.21 shows the CSG-domain KM images.

A total of 251 CSGs are encoded into 51 p_y ray parameters with a range of -0.11 ms/m and 0.11 ms/m. Figures 5.22-5.24 show, respectively, the images of CWKM, CWLSM and RCWLSM (both CWLSM and RCWLSM are stopped after 10 iterations). Compared to CWKM, both CWLSM and RCWLSM can provide high quality images. Figures 5.25-5.26 show the zoom views of the blue and green boxes in Figures 5.22-5.24. Both zoom views show that CWLSM and RCWLSM can produce higher resolution images than CWLSM, but with a high level of noise. CWLSM can provide more continuous reflectors than RCWLSM, which indicates that a simple stacking will blur the final images calculated from RXWLSM. However, the trim statics technique (Huang et al., 2014) fails to fix this problem due to the very high level noise in the prestack images. This indicates the sensitivity to the noise level of the trim statics technique. A 20% computational and IO savings is achieved by the CWLSM and RCWLSM compared to standard LSM.

5.4 Conclusions

The plane-wave least-squares migration algorithms are expanded from 2D to 3D. The computation and IO costs are significantly reduced. A cylindrical-wave encoding LSM method is introduced if the crossline source sampling is too sparse in this direction. The finite-difference solvers to the 3D eikonal equation with plane- and cylindrical-wavefronts are also introduced for calculating the associated travel times. Both synthetic and field results show the improved quality of 3D PWLSM and CWLSM images compared to standard migration images. Field results also show that trim static technique is sensitive to the noise level of prestack images.

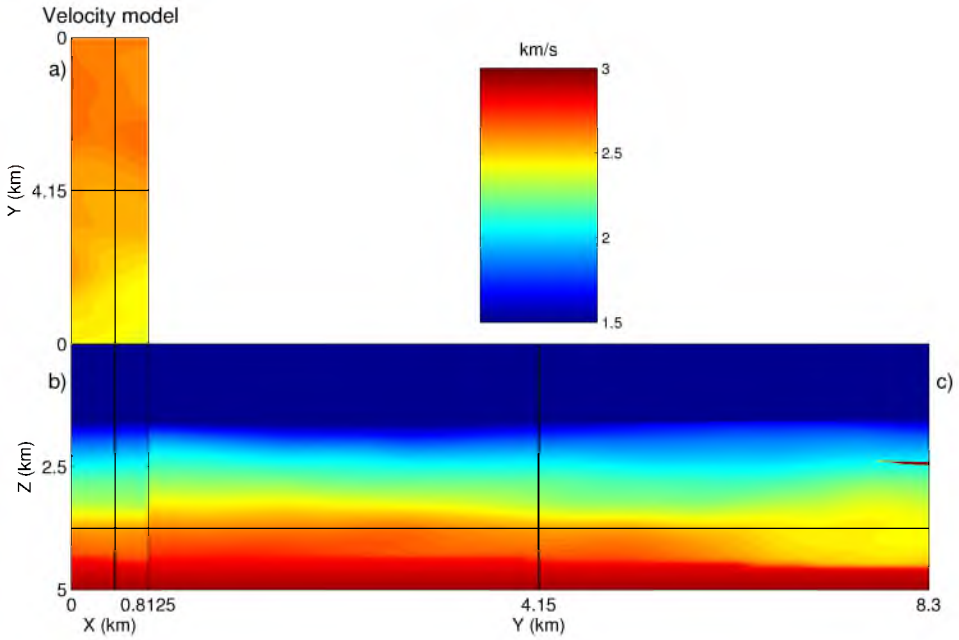

Figure 5.20. Velocity model of GOM data.

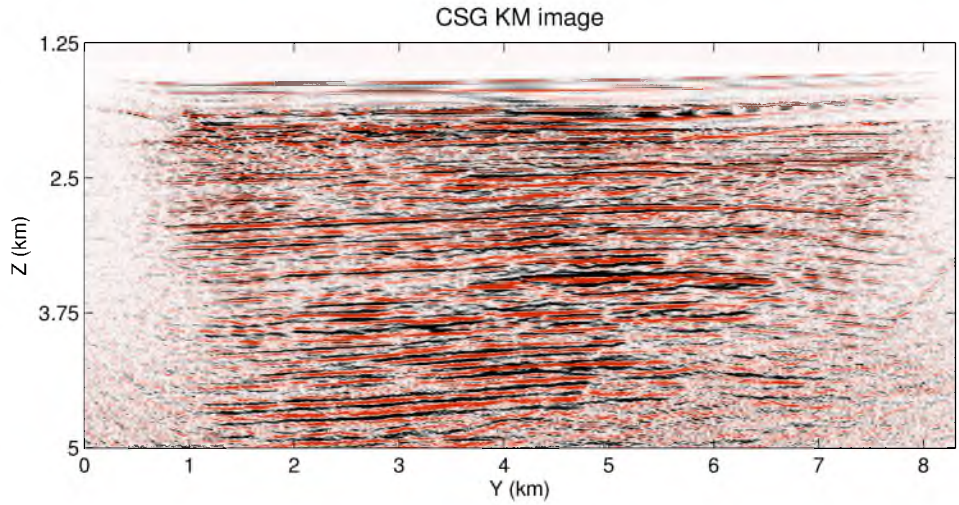

Figure 5.21. CSG KM image of GOM data.

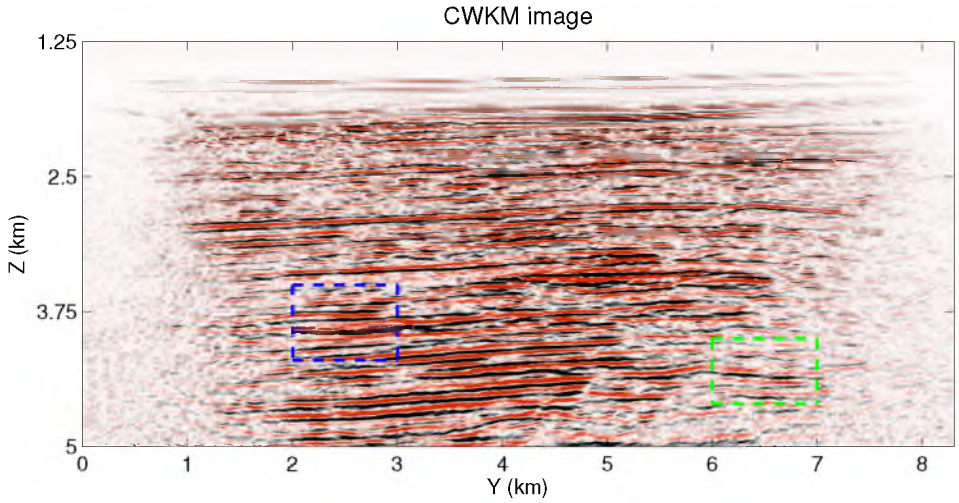

Figure 5.22. CWKM image of GOM data.

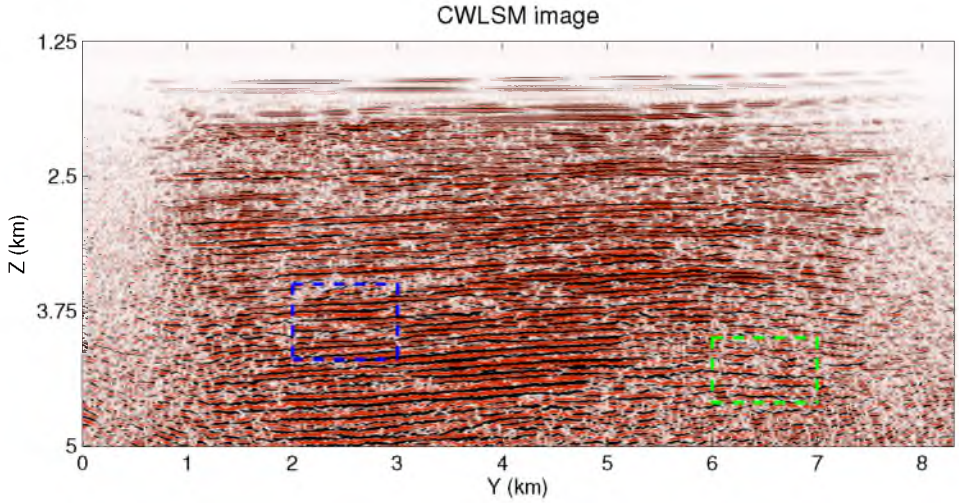

Figure 5.23. CWLSM image of GOM data.

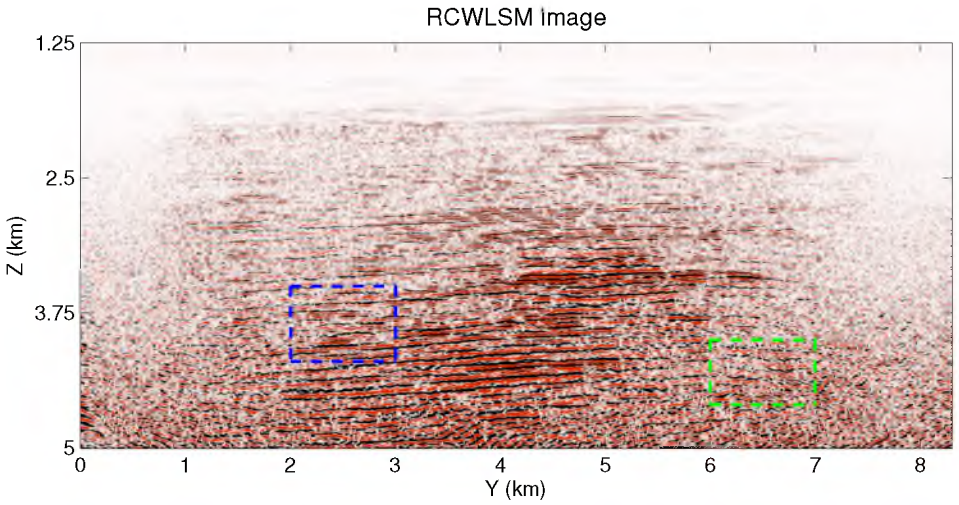

Figure 5.24. RCWLSM image of GOM data.

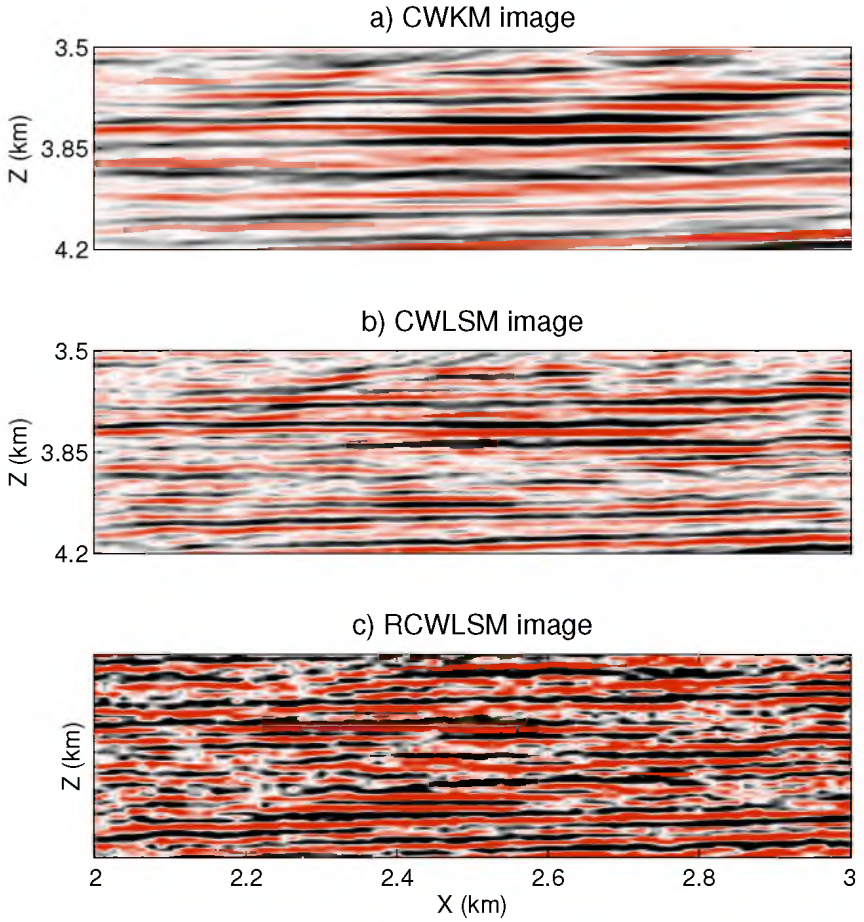

Figure 5.25. Zoom views of the blue boxes in Figures 5.22-5.24.

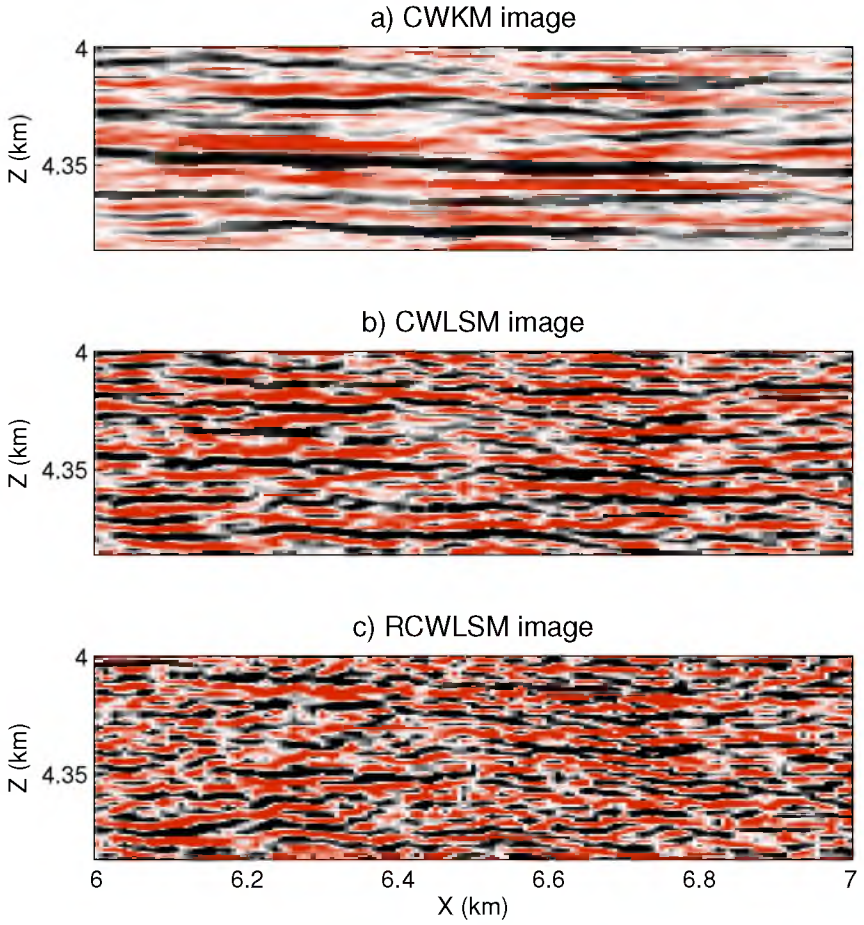

Figure 5.26. Zoom views of the green boxes in Figures 5.22-5.24

APPENDIX A

TRIM STATICS AND LSM

If the common reflectors in two prestack migration images are misaligned in depth by more than a half-wavelength for different shot gathers, then adding them together will lead to a smeared image. This will be true for both the overdetermined (see equation 4.8) and the underdetermined (see equation 4.12) migration images. To alleviate this problem, a trim statics shift can be used to align the prestack migration images with one another prior to stack. The benefit is a more coherent migration image in the presence of inaccurate migration velocities, but the liability might be unacceptable positioning of the reflector image.

One such alignment procedure is an adaptive stacking method. Consider misaligned reflector images in the two prestack migration images denoted as $m(x,z)^A$ and $m(x,z)^B$. The common reflector in both images should be coherently stacked together, so an arbitrary pixel at (x,z) in $m(x,z)^A$ should coherently stack with its 'most similar' counterpart in $m(x,z)^B$ near a small neighborhood around (x,z). This compares to conventional stacking of the event at (x,z) in both the $m(x,z)^A$ and $m(x,z)^B$ images. For a pair of pixels at (x,z) and $(x+\delta x, z+\delta z)$, the similarity based on the pixel values of (x,z) and $(x+\delta x, z+\delta z)$ alone is generally not reliable. A more telling similarity is based on small image patches, of the extent of a few wavelengths, surrounding (x,z) and $(x+\delta x, z+\delta z)$. A pairwise stacking, between images $m(x,z)^A$ and $m(x,z)^B$, is selected when each pixel is stacked with its best counterpart[1]. A 4-fold stacking, among images $m(x,z)^A$, $m(x,z)^B$, $m(x,z)^C$ and $m(x,z)^D$, is achieved by two levels of pairwise stacking, e.g., stack $(m(\mathbf{x})^A, m(\mathbf{x})^B)$ together and stack $(m(\mathbf{x})^C, m(\mathbf{x})^D)$ together, and then stack the two stacked images together. This generalizes to stacking any number of images, termed 'stacking by trim statics.' This stacking procedure can be carried out after all the prestack LSM images are computed, or it can be done after

[1]The best counterpart is defined to be the patch in $m(x+\delta x, z+\delta z)^B$ that gives the most energetic sum after stacking into the patch at $m(x,z)^A$.

every few iterations. The results shown in Figure A.1 were obtained after the prestack LSM images were computed (Huang et al., 2014).

Figures A.1a and A.1b contrast the two methods of stacking produced by least squares Kirchhoff plane-wave migration of plane-wave gathers for 31 different incident angles (Wang et al., 2013). In Figure A.1a there are reflectors truncated laterally, while the one with trim statics in Figure A.1b has much better lateral continuity of reflectors. It is quite evident that, for this example, the latter image is superior to the former in terms of interface continuity.

The liability of trim statics is the danger that the interfaces are broken by actual faulting and the statics corrections destroys this valuable information. In this case, the statics shifts can be used to estimate the depth shift between common reflectors in different CIGs, and this depth residual can be used to update the velocity model (Stork and Clayton, 1986; Stork, 1991, 1992). The trim statics LSM image in Figure A.1b can be used as the *pilot* image from which the depth residuals can be computed for the other CIGs.

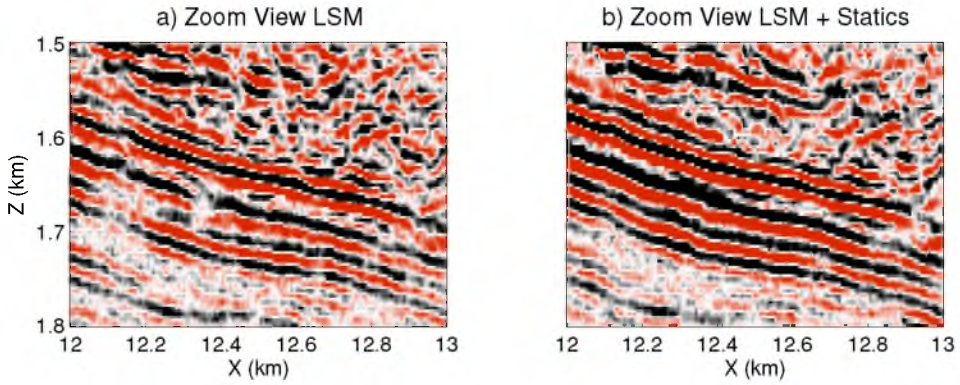

Figure A.1. Comparison of conventional LSM and LSM + trim statics images after 10 iterations of LSM for 31 p-values: a) conventional LSM and b) LSM + trim statics. The data set is recorded from a marine seismic survey in the Gulf of Mexico.

APPENDIX B

3D FINITE DIFFERENCE STENCILS FOR EIKONAL SOLVER

Three finite-difference stencils (illustrated in Figure B.1) for solving the 3D eikonal equation are introduced as:

1. the wavefront at time T_{A1} propagates until the time T_{A6}, where

$$T_{A6} = \begin{cases} T_{A1} + \sqrt{\Delta}, & if \Delta \geq 0, \\ T_{A1} + \sqrt{2}h\bar{s}, & if \Delta < 0, \end{cases} \tag{B.1}$$

$$\Delta = 2h^2\bar{s}^2 - \frac{1}{4}((T_{A2} - T_{A3})^2 + (T_{A4} - T_{A5})^2), \tag{B.2}$$

$$\bar{s} = \frac{1}{2}(s_{A6} + \frac{1}{5}(s_{A1} + s_{A2} + s_{A3} + s_{A4} + s_{A5})), \tag{B.3}$$

   and h is the grid interval;

2. the wavefront at time T_{B1} propagates until the time T_{B8}, where

$$T_{B8} = \begin{cases} T_{B1} + \frac{1}{\sqrt{2}}\sqrt{\Delta}, & if \Delta \geq 0, \\ T_{B1} + \sqrt{3}h\bar{s}, & if \Delta < 0, \end{cases} \tag{B.4}$$

$$\Delta = 6h^2\bar{s}^2 - ((T_{B2} - T_{B7})^2 + (T_{B5} - T_{B3})^2 + (T_{B4} - T_{B6})^2), \tag{B.5}$$

$$\bar{s} = \frac{1}{8}(s_{B1} + s_{B2} + s_{B3} + s_{B4} + s_{B5} + s_{B6} + s_{B7} + s_{B8}); \tag{B.6}$$

3. the wavefront at time T_{C1} propagates until the time T_{C6}, where

$$T_{C6} = \begin{cases} T_{C1} + \frac{1}{\sqrt{2}}\sqrt{\Delta}, & if \Delta \geq 0, \\ T_{C1} + \sqrt{\frac{3}{2}}h\bar{s}, & if \Delta < 0, \end{cases} \tag{B.7}$$

$$\Delta = 2h^2\bar{s}^2 - \frac{1}{2}(T_{C5} - T_{C3})^2 - (T_{C4} - T_{C5})^2, \tag{B.8}$$

$$\bar{s} = \frac{1}{4}(\frac{1}{3}(s_{C1} + s_{C2} + s_{C3}) + s_{C4} + s_{C5} + s_{C6}). \tag{B.9}$$

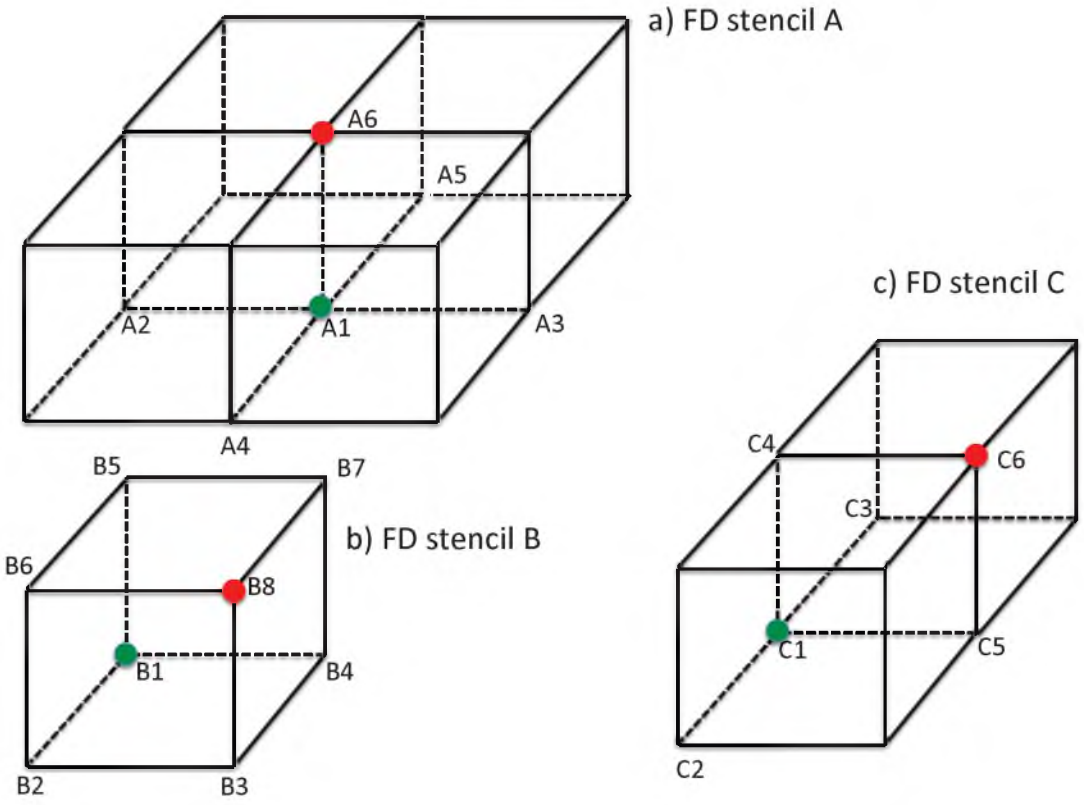

Figure B.1. FD stencils of: a) the wavefront at time T_{A1} propagates until the time T_{A6}, b) the wavefront at time T_{B1} propagates until the time T_{B8}, and c) the wavefront at time T_{C1} propagates until the time T_{C6}. Green-filled circles indicate the timed gridpoints, and the red-filled circles indicate the gridpoints that will be expanded.

REFERENCES

[1] F. Akbar, M. Sen, and P. Stoffa. Prestack plane-wave Kirchhoff migration in laterally varying media. *Geophysics*, 61(4):1068–1079, 1996.

[2] S. Al-Saleh. Application of prestack plane wave Kirchhoff depth migration of obc data. *SEG Technical Program Expanded Abstracts*, 22(1):1089–1092, 2002.

[3] Abdullah Altheyab. Hybrid linear and nonlinear waveform inversion of marine data. *CSIM Annual Report*, pages 1–5, 2012.

[4] Chaiwoot Boonyasiriwat and Gerard T. Schuster. 3D multisource full waveform inversion using dynamic random phase encoding. *SEG Technical Program Expanded Abstracts*, 29(1):1044–1049, 2010.

[5] Chaiwoot Boonyasiriwat, Gerard T. Schuster, Paul Valasek, and Weiping Cao. Applications of multiscale waveform inversion to marine data using a flooding technique and dynamic early-arrival windows. *Geophysics*, 75(6):R129–R136, 2010.

[6] Richard L. Burden and J. Douglas Faires. *Numerical Analysis*. Brooks Cole, 2001.

[7] Vlastislav Červený. *Seismic Ray Theory*. Cambridge University Press, 2005.

[8] N. Chemingui, R. Van Borselen, and M. Orlovich. 3d plane-wave migration of wide-azimuth data. *SEG Technical Program Expanded Abstracts*, 26(1):2195–2199, 2007.

[9] Jon Claerbout. *Earth soundings analysis: Processing versus Inversion*. Blackwell Scientific Publications, 3 Cambridge Center, Cambridge, 1992.

[10] W. Dai and G. Schuster. Plane-wave least-squares reverse-time migration. *Geophysics*, 78(4):S165–S177, 2013.

[11] Wei Dai, C. Boonyasiriwat, and Gerard T. Schuster. 3D multi-source least-squares reverse time migration. *SEG Technical Program Expanded Abstracts*, 29(1):3120–3124, 2010.

[12] Wei Dai, Paul Fowler, and Gerard T. Schuster. Multi-source least-squares reverse time migration. *Geophysical Prospecting*, 60(4):681–695, 2012.

[13] Wei Dai and Gerard T. Schuster. Least-squares migration of simultaneous sources data with a deblurring filter. *SEG Technical Program Expanded Abstracts*, 28(1):2990–2994, 2009.

[14] Wei Dai, Xin Wang, and Gerard T. Schuster. Least-squares migration of multisource data with a deblurring filter. *Geophysics*, 76(5):R135–R146, 2011.

[15] Bertrand Duquet and Patrick Lailly. Efficient 3d wave-equation migration using virtual planar sources. *Geophysics*, 71(5):S185–S197, 2006.

[16] Bertrand Duquet, Kurt J. Marfurt, and Joe A. Dellinger. Kirchhoff modeling, inversion for reflectivity, and subsurface illumination. *Geophysics*, 65(4):1195–1209, 2000.

[17] John T. Etgen. How many angles do we really need for delayed-shot migration? *SEG Technical Program Expanded Abstracts*, 24(1):1985–1988, 2005.

[18] Yunsong Huang and Gerard T. Schuster. Multisource least-squares migration of marine streamer data and land data with frequency-division encoding. *Geophysical Prospecting*, 60(4):663–680, 2012.

[19] Yunsong Huang, Xin Wang, and Gerard T. Schuster. Non-local means filter for trim statics. *SEG Technical Program Expanded Abstracts (submitted)*, 2014.

[20] BR Julian and Db Gubbins. Three-dimensional seismic ray tracing. *J. geophys*, 43(1):95–114, 1977.

[21] Jerome R. Krebs, John E. Anderson, David Hinkley, Ramesh Neelamani, Sunwoong Lee, Anatoly Baumstein, and Martin-Daniel Lacasse. Fast full-wavefield seismic inversion using encoded sources. *Geophysics*, 74(6):WCC177–WCC188, 2009.

[22] Harold J. Kushner and G. George Yin. *Stochastic Approximation and Recursive Algorithms and Applications*. Springer-Verlag, New York, 2003.

[23] Faqi Liu, Douglas W. Hanson, Norman D. Whitmore, Richard S. Day, and Robert H. Stolt. Toward a unified analysis for source plane-wave migration. *Geophysics*, 71(4):S129–S139, 2006.

[24] Peter Mora. Nonlinear two-dimensional elastic inversion of multioffset seismic data. *Geophysics*, 52(9):1211–1228, 1987.

[25] Tamas Nemeth, Chengjun Wu, and Gerard T. Schuster. Least-squares migration of incomplete reflection data. *Geophysics*, 64(1):208–221, 1999.

[26] Athanasios Papoulis and S. Unnikrishna Pillai. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill Companies, 1991.

[27] F. Qin, Y. Luo, K. Olsen, W. Cai, and G. Schuster. Finite-difference solution of the eikonal equation along expanding wavefronts. *Geophysics*, 57(3):478–487, 1992.

[28] Louis A. Romero, Dennis C. Ghiglia, Curtis C. Ober, and Scott A. Morton. Phase encoding of shot records in prestack migration. *Geophysics*, 65(2):426–436, 2000.

[29] Gerard T. Schuster, Xin Wang, Yunsong Huang, Wei Dai, and Chaiwoot Boonyasiriwat. Theory of multisource crosstalk reduction by phase-encoded statics. *Geophysical Journal International*, 184:1289–1303, 2011.

[30] J. Sethian and A. Popovici. 3-D traveltime computation using the fast marching method. *Geophysics*, 64(2):516–523, 1999.

[31] P. Stoffa, M. Sen, R. Seifoullaev, R. Pestana, and J. Fokkema. Plane-wave depth migration. *Geophysics*, 71(6):S261–S272, 2006.

[32] C. Stork. High resolution SVD analysis of the coupled velocity determination and reflector imaging problem. *SEG Technical Program Expanded Abstracts*, 10(1):981–985, 1991.

[33] C. Stork. Reflection tomography in the postmigrated domain. *Geophysics*, 57(5):680–692, 1992.

[34] C. Stork and R. Clayton. Analysis of the resolution between ambiguous velocity and reflector position for traveltime tomography. *SEG Technical Program Expanded Abstracts*, 5(1):545–550, 1986.

[35] Christof Stork and Jerry Kapoor. How many p values do you want to migrate for delayed shot wave equation migration? *SEG Technical Program Expanded Abstracts*, 23(1):1041–1044, 2004.

[36] Albert Tarantola. Linearized inversion of seismic reflection data*. *Geophysical Prospecting*, 32(6):998–1015, 1984.

[37] Jos van Trier and William W Symes. Upwind finite-difference calculation of travel-times. *Geophysics*, 56(6):812–821, 1991.

[38] John Vidale. Finite-difference calculation of travel times. *Bulletin of the Seismological Society of America*, 78(6):2062–2076, 1988.

[39] Denes Vigh and E. William Starr. 3D prestack plane-wave, full-waveform inversion. *Geophysics*, 73(5):VE135–VE144, 2008.

[40] J. Virieux and S. Operto. An overview of full-waveform inversion in exploration geophysics. *Geophysics*, 74(6):WCC1–WCC26, 2009.

[41] Jean Virieux and Veronique Farra. Ray tracing in 3-d complex isotropic media: An analysis of the problem. *Geophysics*, 56(12):2057–2069, 1991.

[42] Juefu Wang and Mauricio D. Sacchi. High-resolution wave-equation amplitude-variation-with-ray-parameter (avp) imaging with sparseness constraints. *Geophysics*, 72(1):S11–S18, 2007.

[43] X. Wang, W. Dai, and G. Schuster. Regularized plane-wave least-squares kirchhoff migration. *SEG Technical Program Expanded Abstracts*, 24(1):278–291, 2013.

[44] Xin Wang and Gerard T. Schuster. Multi-source least-squares migration of marine data. *SEG Technical Program Expanded Abstracts*, 31(1):1–5, 2012.

[45] Yu Zhang, James Sun, Carl Notfors, Samuel H. Gray, Leon Chernis, and Jerry Young. Delayed-shot 3D depth migration. *Geophysics*, 70(5):E21–E28, 2005.

[46] Hongkai Zhao. A fast sweeping method for eikonal equations. *Mathematics of computation*, 74(250):603–627, 2005.